

General analysis system

Patent number: DE69328784T
Publication date: 2000-11-16
Inventor: NORELL LENNART (SE)
Applicant: ERICSSON TELEFON AB L M (SE)
Classification:
- international: H04Q3/545; G06F9/44
- european: G06F9/44; H04Q3/545T2
Application number: DE19936028784T 19931026
Priority number(s): WO1993SE00882 19931026; US19920974165 19921110

Also published as:

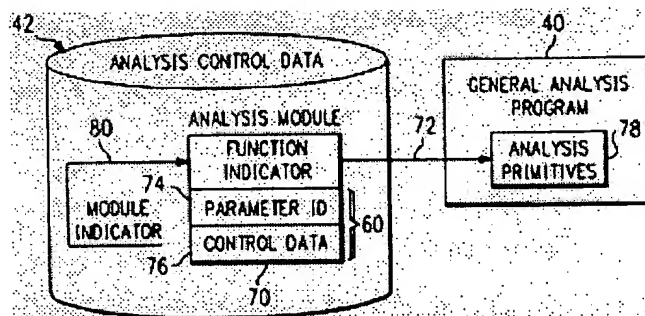
WO9411996 (A1)
EP0669071 (A1)
US6134304 (A1)
FI952252 (A)
EP0669071 (B1)

more >>

[Report a data error here](#)

Abstract not available for DE69328784T
Abstract of corresponding document: **US6134304**

A general analysis system for performing an analysis used in the control of an industrial process. The general analysis system comprises a general analysis program for performing at least one analysis on a set of data and an analysis control data for controlling the analysis of the data set. According to the present invention, the general analysis program comprises a plurality of analysis primitives and the analysis control data comprises a plurality of analysis modules. An analysis may be constructed from a set of analysis modules and performed by interpreting this set using at least one of the analysis primitives.



Data supplied from the **esp@cenet** database - Worldwide

BEST AVAILABLE COPY



DEUTSCHES
PATENT- UND
MARKENAMT

12 Übersetzung der
europäischen Patentschrift

97 EP 0 669 071 B 1

10 DE 693 28 784 T 2

51 Int. Cl. 7:
H 04 Q 3/545
G 06 F 9/44

- | | | |
|----|---|----------------|
| 21 | Deutsches Aktenzeichen: | 693 28 784.5 |
| 66 | PCT-Aktenzeichen: | PCT/SE93/00882 |
| 96 | Europäisches Aktenzeichen: | 93 924 856.3 |
| 87 | PCT-Veröffentlichungs-Nr.: | WO 94/11996 |
| 86 | PCT-Anmeldetag: | 26. 10. 1993 |
| 87 | Veröffentlichungstag
der PCT-Anmeldung: | 26. 5. 1994 |
| 97 | Erstveröffentlichung durch das EPA: | 30. 8. 1995 |
| 97 | Veröffentlichungstag
der Patenterteilung beim EPA: | 31. 5. 2000 |
| 47 | Veröffentlichungstag im Patentblatt: | 16. 11. 2000 |

- 30 Unionspriorität:
974165 10. 11. 1992 US
- 73 Patentinhaber:
Telefonaktiebolaget L M Ericsson (publ),
Stockholm, SE
- 74 Vertreter:
HOFFMANN · EITLE, 81925 München
- 84 Benannte Vertragsstaaten:
BE, CH, DE, DK, ES, FR, GB, GR, IE, IT, LI, NL, SE

- 72 Erfinder:
NORELL, Lennart, Bror, S-125 30 Älvsjö, SE

54 ALLGEMEINES ANALYSESYSTEM

Anmerkung: Innerhalb von neun Monaten nach der Bekanntmachung des Hinweises auf die Erteilung des europäischen Patents kann jedermann beim Europäischen Patentamt gegen das erteilte europäische Patent Einspruch einlegen. Der Einspruch ist schriftlich einzureichen und zu begründen. Er gilt erst als eingelegt, wenn die Einspruchsgebühr entrichtet worden ist (Art. 99 (1) Europäisches Patentübereinkommen).

Die Übersetzung ist gemäß Artikel II § 3 Abs. 1 IntPatÜG 1991 vom Patentinhaber eingereicht worden. Sie wurde vom Deutschen Patent- und Markenamt inhaltlich nicht geprüft.

DE 693 28 784 T 2

DF 693 28 784 T 2

HINTERGRUND DER ERFINDUNG

Gebiet der Erfindung

Die vorliegende Erfindung betrifft eine allgemeines Analysesystem zum Ausführen einer diversen Analyse von Information in einer Vielzahl von Einrichtungen, einschließlich insbesondere einer Telefonvermittlungsstelle.

Geschichte des Standes der Technik

Grob gesagt umfasst die vorliegende Erfindung einen neuartigen Ansatz für die Analyse von Daten, der ersetzt, was bislang erzielt worden ist durch Implementieren jedes Analysealgorithmus in einem analyse-spezifischen Computerprogramm. Anstelle der Erzeugung von verschiedenen Computerprogrammen für verschiedene Typen von Analysen, so wie dies im Stand der Technik herkömmlich ist, erwägt der neuartige Ansatz der vorliegenden Erfindung die Erzeugung eines generischen Analyseprogramms und der Beschreibung der Unterschiede zwischen Analysen in den Daten, mit denen das generische Analyseprogramm arbeitet. Dieser Ansatz kann deshalb als daten-geführt anstelle von software-geführt mit einer eingebauten Analyse charakterisiert werden. Die Daten können von dem Benutzer in einem einfachen Format definiert werden, das im Gegensatz zu Computerprogrammiersprachenformaten eine geringe oder keinerlei Expertise, Kenntnis, Training oder Fertigkeiten erfordert.

Aus der nun folgenden ausführlichen Beschreibung lässt sich ersehen, dass der neuartige Ansatz der vorliegenden Erfindung Vorteile in einem breitem Spektrum von industriellen Anwendungen aufweist. Irgendein industrielles

Prozesssteuersystem, das für verschiedene Installationen zugeschnitten werden muss und an jeder Installation periodisch modifiziert werden muss, um neue Eingaben zu dem und Ausgaben von dem Prozess zu berücksichtigen, kann allgemein von den Lehren der vorliegenden Erfindung profitieren. Ein Hauptbeispiel eines derartigen industriellen Prozesses lässt sich im Gebiet der Telekommunikationen und insbesondere bei der Verbindung von Anrufen zwischen Telefonteilnehmern finden. Für Zwecke der folgenden Beschreibung umfasst der Ausdruck "Vermittlungsstelle" oder "Vermittlungsmaschine" wie anwendbar eine Fernverkehrs-Steuervermittlungsstelle, eine speziell vorgesehene zellulare Vermittlungsstelle oder eine lokale Vermittlungsstelle.

Heutzutage wird ein Telefondienst einer Vielzahl von Kunden oder Telefonteilnehmern über eine zentralisierte Vermittlung bereitgestellt. Eine Maschine für eine zentralisierte Vermittlung in einer zentralen Vermittlungsstelle steuert die Vermittlung der Anrufe zu den und von den lokalen Telefonteilnehmern und kommuniziert mit anderen zentralen Vermittlungsstellen in dem Netz über Bündelleitungen. Jede zentrale Vermittlungsstelle muss bestimmte Funktionen beim Behandeln eines einfachen Anrufs ausführen. Zum Beispiel muss die zentrale Vermittlungsstelle überwachen und erfassen, dass ein Kunde einen Dienst wünscht, wenn das Telefon des Kunden abgehoben wird und der Kunde einen Anruf einleitet. Sobald die zentrale Vermittlungsstelle erkennt, dass eine Einleitung vorgenommen worden ist, d.h. den abgehobenen Status einer gegebenen Leitung erfasst, muss die zentrale Vermittlungsstelle eine Verbindung zu der Leitungseinrichtung herstellen, um den Kunden z.B. über einen Wählton darüber zu benachrichtigen, dass die zentrale Vermittlungsstelle bereit ist, und eine Einrichtung zum Empfangen dieser Information. Information, z.B. die angerufene Nummer, wird von dem Kunden über eine Drehscheibe oder eine Tastatur eingegeben und wird an der zentralen Vermittlungsstelle empfangen und aufgezeichnet. Diese Information muss dann von den Geräten der

zentralen Vermittlungsstelle interpretiert werden, um die Stelle der angerufenen Leitung zu identifizieren.

Wenn die angerufene Partei und die anrufende Partei von der gleichen zentralen Vermittlungsstelle bedient werden, d.h. der Anruf ein Intravermittlungsstellenanruf ist, wird ein Belegtest der angerufenen Leitung durchgeführt und wenn die angerufene Leitung frei ist, dann wird die angerufene Partei alarmiert, z.B. über einen hörbaren Klingelton angerufen. Die angerufene Leitung wird überwacht, wobei eine Antwort von der angerufenen Partei oder ein Abbruch von der anrufenden Partei abgewartet wird. Wenn die angerufene Partei antwortet, wird ein Sprachpfad zwischen den Parteien hergestellt. Der Sprachpfad wird während einer Konversation überwacht und wird getrennt, wenn eine der Parteien auflegt und die Leitung in den aufgelegten Zustand geht.

Wenn andererseits die angerufene Partei von einer anderen zentralen Vermittlungsstelle bedient wird, d.h. der Anruf ein Intervermittlungsstellenanruf ist, dann wird eine Suche nach einer freien direkten Bündelleitung zu der zentralen Vermittlungsstelle, die die angerufene Partei bedient, oder zu einer dazwischenliegenden zentralen Vermittlungsstelle, die die Weiterleitung des Anrufs zu der zentralen Vermittlungsstelle der angerufenen Partei unterstützen kann, durchgeführt. Information über die angerufene Nummer wird von der zentralen Vermittlungsstelle, die die Einleitung vornimmt, gesendet und von der dazwischenliegenden zentralen Vermittlungsstelle empfangen, die die Information bis zu der abschließenden (Empfangs-) zentralen Vermittlungsstelle liefert. Wenn die Leitung der angerufenen Partei belegt ist oder der Anruf irgendwo in dem Netz abgeblockt wird oder die erforderlichen Intervermittlungsstellen-Bündelungsleitungen alle belegt sind, wird der anrufende Teilnehmer über einen hörbaren belegt, fast belegt oder Rekorder-Ton informiert.

Die Vermittlungsfunktionen, die zur Behandlung des voranstehend illustrierten einfachen Telefonanrufs vorhanden sind, können grob als Signalempfang, Signalinterpretation, Speicherung, Pfadauswahl, Netzpfadbereitstellung und Steuerung und Signalübertragung klassifiziert werden. Zusätzlich zu diesen Kundendienstfunktionen muss das Vermittlungssystem eine Verwendung registrieren und vielleicht sogar auf einer periodischen Basis die Gebühren in Zusammenhang mit der Verwendung, die von einer bestimmten Rufnummer ausgeht, berechnen. Moderne Vermittlungsmaschinen führen wenigstens eine von diesen und andere Funktionen unter einer elektronischen Steuerung (verdrahtete Logik oder gespeichertes Programm) aus. Insbesondere werden auf Telefon-Vermittlungsmaschinen zunehmend die Konzepte einer Echtzeitverarbeitung, d.h. eines Betriebs bei einer ausreichend hohen Rate, um sich schnell verändernde Ereignisse zu verfolgen, und eines interaktiven Betriebs, d.h. der Fähigkeit, mit der Information von einem Menschen auf einer Multitransaktionsbasis in Wechselwirkung zu treten und diese Information anzunehmen und zu verarbeiten, die traditionelle zu Computersystemen gehören, angewendet.

Fig. 1 zeigt schematisch einige der vielen Analysefunktionen, die beim Herstellen von Kommunikation zwischen zwei Telefonteilnehmern A und B ausgeführt werden. Der Teilnehmer A verwendet eine Telefoneinheit 10, um an einen Teilnehmer B einen Anruf zu richten, und ist mit einer lokalen Vermittlungsstelle 12 verbunden. Die lokale Vermittlungsstelle 12 erfasst die Anforderung nach einem Dienst, d.h. die Ankunft der gewählten Ziffern oder Zahlen, von dem A-Teilnehmer und führt eine Anzahl von Analyseprozeduren (Funktionen) aus, einschließlich einer Dienstanalyse (SA) 14, einer Zielstellenanalyse (DA) 16 und einer Verzweigungs (Routing)-Analyse (RA) 18 in dieser Reihenfolge. Während zur Vereinfachung nur drei Analysefunktionen SA 14, DA 16 und RA 18 in Fig. 1 gezeigt sind, können mehrere weitere Analysefunktionen in der lokalen Vermittlungsstelle 12

ausgeführt werden, um den Anruf von dem A-Teilnehmer zu dem B-Teilnehmer aufzubauen.

Der Zweck der SA 14 besteht darin, den anrufenden Teilnehmer (in diesem Fall den A-Teilnehmer) zu identifizieren und zu überprüfen, ob der anrufende Teilnehmer aufgrund seines oder ihres Teilnahmeberechtigungsstatus und anderer wichtiger Überlegungen berechtigt ist, einen Anruf durchzuführen. Die Rufnummer des A-Teilnehmers kann für diese Zwecke überprüft werden. Der Hauptzweck der DA 16 besteht darin, die Zielstelle zu bestimmen, an der der B-Teilnehmer mit einem Netz verbunden ist, d.h. den Ort der Telefoneinheit 24. Für diesen Zweck wird die gewählte Nummer (die Telefonnummer des B-Teilnehmers) in Anbetracht des Aufenthaltsorts des A-Teilnehmers (gleiche Vermittlungsstelle oder eine andere Vermittlungsstelle), ob der B-Teilnehmer zu einer Centrex-Gruppe (Centrex ist ein Dienst, der mit Merkmalen einer PBX vergleichbar ist, aber mit einem Teil oder der gesamten Steuerung in der zentralen Vermittlungsstelle implementiert) gehört oder nicht, und anderer wichtiger Faktoren interpretiert.

Das Ziel der RA 18 besteht darin, eine Netzroute zu lokalisieren, durch die der Anruf von dem A-Teilnehmer mit dem B-Teilnehmer verbunden werden kann. Diese Analyse hängt nicht nur von der abschließenden Zielstelle des Anrufs, d.h. des Aufenthaltsorts des B-Teilnehmers, ab, sondern auch von einem Verkehrsstau in dem Netz, d.h. der Last auf verschiedenen Routen durch das Netz, der Art des Anrufs, d.h. analog oder digital, und ähnlichen Anzeichen über die Netzkapazität. Auf Grundlage der Ergebnisse der RA 18 kann die lokale Vermittlungsstelle 12 den Anruf an eine zwischenliegende (Fern- oder Transit-) Vermittlungsstelle 20 leiten, die wiederum den Anruf durch eine lokale Vermittlungsstelle 22 zu der Telefoneinheit 24 des B-Teilnehmers leitet.

Fig. 2 zeigt eine Steuerung mit einem gespeicherten Programm der Analysefunktionen, die in der lokalen Vermittlungsstelle

12 der Fig. 1 ausgeführt werden. Gemäss Fig. 2 wird jede Analyse, z.B. SA 14, DA 16 und RA 18 mit einem anderen Analyseprogramm 30 implementiert. Der Eingang an jedem Analyseprogramm 20 umfasst Anrufparameter, z.B. eine A-Teilnehmer-Nummer und einen Ortscode, eine B-Teilnehmernummer und einen Ortscode, etc., die aus einem vorgegebenen Feld von Parametern 22 zugeführt werden. Die Parameter werden unter Bezugnahme auf eine Datenstruktur 34 analysiert, die für jeden Typ von Analyse spezifiziert ist und die Kombinationen von Parameterwerten interpretiert und ein Analyseergebnis erzeugt. Jedes Analyseprogramm 30 und jede zugehörige Datenstruktur 34 kann von dem Netzbetreiber über eine Mensch-Maschine-Schnittstelle 36 gesteuert werden. Insbesondere kann der Netzbetreiber geeignete Werte auf die Daten in jeder Datenstruktur 34 anwenden, so dass die verwandte Analyse ein gewünschtes Ergebnis bei gegebenem Aufenthaltsort der lokalen Vermittlungsstelle in dem Netz, dem Nummerierungsplan etc. erzeugen wird. Z.B. kann der Betreiber wählen, die Vorgehensweise zu ändern, in der Anrufe an eine bestimmte Zielstelle innerhalb des Netzes geroutet werden, indem geeignete Werte auf die Daten in der Datenstruktur 34 im Zusammenhang mit der RA 18 angewendet werden.

Der in Fig. 2 illustrierte herkömmliche Analyseansatz ist ähnlich zu dem Ansatz, der für die Analyse von Raumleistungssystemen in Lilley, J.R., "A Code for Conducting Systems-Level Survivability Analysis of Nuclear Power Systems", Proceedings of the 26th InterSociety Energy Conversion Engineering Conference, August 1991 (IEE-91), Seiten 201-205), offenbart ist. Dieser Typ von Ansatz weist eine Anzahl von Nachteilen und Unzulänglichkeiten auf. Weil jede unterschiedliche und spezialisierte Analyse insbesondere ein spezifisches Analyseprogramm 30 erfordert, muss ein spezialisierter (Analyse-spezifischer) Satz von Daten, die in einer spezifischen Struktur 34 konfiguriert werden, erstellt und über eine Mensch-Maschine-Schnittstelle 36 für jedes Analyseprogramm 30 eingegeben werden. Zum Beispiel muss für

jeden dieser Datensätze spezifisch eine Datenstruktur 34, die den Nummerierungsplan definiert, der in der DA 16 verwendet werden soll, und eine andere Datenstruktur 34, die die gewünschten Wegleitungsmuster (Routing-Master) in dem Bündelungsnetz definiert, die in der RA 18 verwendet werden sollen, erstellt werden. Ein beträchtlicher Aufwand an Programmierzeit und Anstrengungen von höchst erfahrenem Personal muss deshalb beim Erzeugen und Entwickeln jeder unterschiedlichen Analyse und des Gesamtanalysepakets investiert werden. Ferner ist der Typ von Eingangsparametern zu und die Art von Ausgangsergebnissen von jeder Analyse alle vorgegeben. Demzufolge bedingt die Einführung von neuen Typen von Parametern oder Ergebnissen entweder die Neuprogrammierung von existierenden Versionen oder die Freigabe von neuen Versionen des (der) relevanten Analyseprogramms (-programme) 30 und der Datenstruktur (Datenstrukturen) 34, alle bei beträchtlichen Ausgaben.

In der Praxis ist es jedoch während eines tatsächlichen Betriebs oft erforderlich und wünschenswert, eine bestimmte Analysefunktion zu modifizieren, um zum Beispiel den Verkehr neu zu richten, wenn das Netz überlastet ist, um verschiedene Parameter wie das Volumen oder die Dichte eines Verkehrs durch einen bestimmten Teil des Netzes zu messen, um zusätzliche Parameter in die Analyse einzubauen, usw. Gemäss der herkömmlichen Analysesysteme, wie dem in Fig. 2 dargestellten System, kann jedoch der Netzbetreiber und/oder der Vermittlungsstellenanbieter mit der im Grunde genommen unmöglichen Aufgabe belastet werden, sämtliche zukünftigen Modifikationen an dem Netz oder an einer bestimmten Vermittlungsstelle in dem Netz vorzusehen oder wenigstens mit der häufigen Aufgabe, eine Integration eines neuen Codes mit einem existierenden Code und ein Debugging der sich ergebenden Kombination.

Angesichts dieser Nachteile und Unzulänglichkeiten der herkömmlichen Systeme ist es eine Aufgabe der vorliegenden

Erfindung, ein allgemeines Analysesystem bereitzustellen, welches nicht für irgendeine einzelnen Analyse spezifisch ist und welches zum Ausführen einer großen Bandbreite von Analysen unabhängig von den Eigenheiten jeder ausgeführten Analyse verwendet werden kann. Das allgemeine Analysesystem kann als ein generisches Analysesystem anstelle der extensiven und relativ teuren Sätze von spezialisierten Programmen und Datenstrukturen von herkömmlichen Systemen verwendet werden.

Es ist eine andere Aufgabe der vorliegenden Erfindung, ein allgemeines Analyseprogramm und eine Analysesteuerdatenstruktur innerhalb des allgemeinen Analysesystems bereitzustellen, die zum Ausführen irgendeiner gewünschten Analyse verwendet werden können.

Es ist noch eine andere Aufgabe der vorliegenden Erfindung, ein allgemeines Analysesystem bereitzustellen, bei dem neue Funktionen definiert werden können in und neue Parameter zum Beeinflussen der Analyse vorgesehen sind, angezeigt werden können aus den Analysesteuerdaten. Neue Funktionen und neue Parameter können zu einer Analyse hinzugefügt werden, ohne das allgemeine Analysesystem erneut zu programmieren, indem Betreiber-zugeführte Daten in dem allgemeinen Analyseprogramm oder Kunden-zugeführte Daten in den Analysesteuerdaten verwendet werden, was die Grundlage der Analyse bildet.

Es ist eine weitere Aufgabe der vorliegenden Erfindung, eine Anrufsteuerdatenstruktur bereitzustellen, die Eingangsparameter und Ausgangsergebnisse speichert und die ferner ermöglicht, dass Parameter von den Analysesteuerdaten angezeigt werden können.

ZUSAMMENFASSUNG DER ERFINDUNG

Die Struktur und der Betrieb der vorliegenden Erfindung werden nachstehend ausführlich beschrieben. Kurz zusammengefasst stellt die vorliegende Erfindung jedoch ein allgemeines

Analysesystem zum Ausführen einer Analyse bereit, die bei der Steuerung eines industriellen Prozesses verwendet wird. Das allgemeine Analysesystem umfasst ein allgemeines Analyseprogramm, um wenigstens eine Analyse eines Satzes von Daten und eines Analysedatenwerts zum Steuern der Analyse des Datensatzes auszuführen. Gemäss der vorliegenden Erfindung umfasst das allgemeine Analyseprogramm eine Vielzahl von Analyse-Primitiven (Analysegrundelementen) und die Analysesteuerdaten umfassen eine Vielzahl von Analysemodulen. Eine Analyse kann aus einem Satz von Analysemodulen konstruiert werden und durch Interpretieren dieses Satzes unter Verwendung wenigstens eines der Analyse-Primitive ausgeführt werden (ein Primitiv ist eine Befehlseinheit auf tiefstem Sprachniveau).

In einem anderen Aspekt stellt die vorliegende Erfindung ein System zum Bearbeiten von Telefonanrufen bereit, umfassend wenigstens eine Vermittlungsstelle und in jeder Vermittlungsstelle ein allgemeines Analyseprogramm mit einer Vielzahl von Analyse-Primitiven, Analysesteuerdaten mit einer Vielzahl von Analysemodulen und Anrufsteuerdaten mit einer Vielzahl von Anrufaufzeichnungen, jeweils eine für jeden Telefonanruf durch die Vermittlungsstelle. Die Analysemodule können in einer Vielzahl von Analysestrukturen gruppiert werden, wobei jede von diesen eine andere Analyse steuert. Die Analysestrukturen können z.B. eine Wegleitungs(Routing)analysestruktur und eine Zielstellenanalysestruktur umfassen.

In einem noch anderen Aspekt stellt die vorliegende Erfindung ein Verfahren zum Steuern eines industriellen Prozesses bereit, umfassend die Schritte zum Speichern einer Vielzahl von Parametern für den industriellen Prozess, Ausführen einer Vielzahl von Analysemodulen, wobei jedes davon einen Teil einer Analyse der Parameter steuert und wobei wenigstens eines davon ein Ergebnis aus der Analyse erzeugt, und Steuern des industriellen Prozesses auf

Grundlage des erzeugten Ergebnisses. Die Parameter können eine Eingabe umfassen, die aus dem industriellen Prozess erzeugt wird, Daten, die Charakteristiken des industriellen Prozesses definieren, und das (die) Ergebnis(se) von früher ausgeführten Analysen. Das Ergebnis kann ein Zwischenergebnis oder ein Endergebnis aus der Analyse sein.

In einem weiteren Aspekt stellt die vorliegende Erfindung ein Verfahren zum Sammeln und Ausführen wenigstens einer Analyse bereit, die zum Steuern eines industriellen Prozesses verwendet wird, umfassend die Schritte zum Bereitstellen einer Vielzahl von High-Level-Analyseprimitiven, die jeweils eine Bildungs-Prozedur und eine Do-Prozedur umfassen, ein Zusammenstellen (Kompilieren) einer Vielzahl von Steuerebenen-Analysemodule unter Verwendung der Bildungs-Prozedur von wenigstens einem der High-Level-Analyseprimitive, wobei die Steuerebenen-Analysemodule eine Baumstruktur bilden, die eine Analyse darstellt, und Ausführen jedes Steuerebenen-Analysemoduls in der Baumstruktur unter Verwendung der Do-Prozedur des High-Level-Analyseprimitivs, das zum Zusammenstellen (Kompilieren) des Steuerebenen-Analysemoduls verwendet wurde.

KURZBESCHREIBUNG DER ZEICHNUNGEN

Die vorliegende Erfindung lässt sich besser verstehen und ihre zahlreichen Aufgaben und Vorteile ergeben sich einem Durchschnittsfachmann deutlicher unter Bezugnahme auf die beiliegenden Zeichnungen, in denen gleiche Bezugszeichen oder Zahlen die gleichen Objekte bezeichnen. In den Zeichnungen zeigen:

Fig. 1 eine schematische Darstellung von einigen der Analysefunktionen, die von einer lokalen Vermittlungsstelle bei dem Prozess einer Herstellung von Kommunikationen zwischen zwei Telefonteilnehmern A und B ausgeführt werden;

- Fig. 2 eine schematische Darstellung der herkömmlichen Speicherprogrammsteuerung der Analysefunktionen, die in der in Fig. 1 gezeigten lokalen Vermittlungsstelle ausgeführt werden;
- Fig. 3 ein vereinfachtes schematisches Blockschaltbild des allgemeinen Analysebilds der vorliegenden Erfindung;
- Fig. 4 ein erweitertes schematisches Blockschaltbild des allgemeinen Analysesystems, das in vereinfachter Form in Fig. 3 gezeigt ist;
- Fig. 5 eine bildliche Darstellung der Konstruktion des allgemeinen Analyseprogramms und der Analysesteuerdaten, die in den Fig. 3-4 gezeigt sind;
- Fig. 6 eine bildliche Darstellung der Konstruktion der in Fig. 4 gezeigten Anrufsteuerdaten;
- Fig. 7 eine bildliche Darstellung der Wechselwirkung zwischen dem allgemeinen Analyseprogramm, den Analysesteuerdaten und den Anrufsteuerdaten in einer beispielhaften Analyse, die von dem allgemeinen Analysesystem der vorliegenden Erfindung ausgeführt wird;
- Fig. 8 eine bildliche Darstellung der Konstruktion einer Zielstellenanalyse aus einer Kette von Analysemodulen der Form, die in den Fig. 5-7 gezeigt ist;
- Fig. 9A-B schematische Darstellungen von zwei beispielhaften Verwendungen der Steuerdaten in dem Steuerfeld eines Analysemoduls;

- Fig. 10 eine schematische Darstellung des Aufbaus einer bevorzugten Ausführungsform für das allgemeine Analyseprogramm der vorliegenden Erfindung;
- Fig. 11 eine schematische Darstellung des Aufbaus einer bevorzugten Ausführungsform für die Analysesteuerdaten der vorliegenden Erfindung;
- Fig. 12 eine schematische Darstellung des Aufbaus eines bestimmten Typs von Analysemodulen in der bevorzugten Ausführungsform der in Fig. 11 gezeigten Analysesteuerdaten;
- Fig. 13 eine bildliche Darstellung der Verwendung eines beispielhaften Analysemoduls des Typs, der in Fig. 12 gezeigt ist;
- Fig. 14 eine bildliche Darstellung der Vorgehensweise, mit der ein Analysemodul auf der unteren Ebene (Low-Level) aus Analyseprimitiven der unteren Ebene konstruiert werden kann;
- Fig. 15 eine bildliche Darstellung, wie das vom Benutzer definierte Analysemodul auf der unteren Ebene, das in Fig. 14 gezeigt ist, beim Bilden eines Analysemoduls auf der hohen Ebene (High-Level) verwendet werden kann;
- Fig. 16 eine bildliche Darstellung, wie ein Steuerebenen-Analysemodul, ähnlich wie das in Fig. 12 gezeigte beispielhafte Analysemodul, mit der Bildungs-Prozedur des Analysemoduls der oberen Ebene, das in Fig. 15 gezeigt ist, eingerichtet werden kann; und
- Fig. 17 eine bildliche Darstellung einer Betreiberschnittstelle zu dem allgemeinen Analysesystem der vorliegenden Erfindung.

AUSFÜHRLICHE BESCHREIBUNG

Ein vorläufiger Überblick über einige der besonderen Merkmale der vorliegenden Erfindung kann eine gewisse Führung und Unterstützung beim Verständnis der ausführlichen Beschreibung der bevorzugten Ausführungsform, die folgt, anbieten.

Allgemeiner Überblick

Die Konstruktion des allgemeinen Analysesystems der vorliegenden Erfindung ermöglicht die gesamte Spezifikation einer Analyse innerhalb der Analysesteuerdaten. Mit anderen Worten, die Antworten auf die Fragen, wie analysiert werden soll und wie analysiert werden wird, können von dem Betreiber in bestimmten Elementen (nachstehend als "Analysemodule" bezeichnet) in den Analysesteuerdaten spezifiziert werden. Die Konstruktion der Analysesteuerdatenkomponente des allgemeinen Analysesystems der vorliegenden Erfindung ermöglicht, dass die Mensch-Maschine-Schnittstelle vollständig die Erfordernisse des Betreibers in einer benutzerfreundlichen Weise aufnimmt. Ferner ist die Konstruktion der allgemeinen Analyseprogrammkomponente des allgemeinen Analysesystems der vorliegenden Erfindung grösstenteils unabhängig von dem zu analysierenden Gegenstand und dem beabsichtigten Ergebnis (den beabsichtigten Ergebnissen) der Analyse. Die bestimmten Unterprogramme (die nachstehend als "Analyseprimitive" bezeichnet werden) in dem allgemeinen Analyseprogramm, die in einer Analyse ausgeführt werden sollen, werden durch Analysemodule in den Analysesteuerdaten angezeigt. Die Analysemodule in den Analysesteuerdaten können auch die Ausführung von anderen Unterprogrammen andeuten, die anstelle der Analysesteuerdaten vorhanden sind. Zusätzlich zu der Ausführung von Analysen können die Unterprogramme der vorliegenden Erfindung zum Beispiel Analysedaten von anderen Vermittlungsstellen in dem Netz sammeln oder technische oder statistische Information, die sich auf das Verhalten des

Netzes oder irgendeinen der mit dem Netz verbundenen Telefonteilnehmer bezieht, registrieren.

Unter Bezugnahme nun auf Fig. 3 lässt sich ein vereinfachtes schematisches Blockschaltbild des allgemeinen Analysesystems der vorliegenden Erfindung ansehen. Anstelle einer Vielzahl von Analyseprogrammen und entsprechenden Datenstrukturen beinhaltet das allgemeine Analysesystem ein generisches Analyseprogramm (nachstehend als das "allgemeine Analyseprogramm" bezeichnet) 40, das eine gewünschte Analyse von einer Vielzahl von verfügbaren Analysen unter der Steuerung der Analysesteuerdaten 42 ausführt. Grob gesagt stellen die Analysesteuerdaten 42 einen "Blueprint" zum Steuern des Verhaltens einer bestimmten Analyse eines Satzes von Daten, die für jede Ausführung dieser Analyse unterschiedlich sein können, z.B. verschiedenen Datensätze für verschiedene Telefonanrufe, die in ähnlicher Weise analysiert werden sollen, bereit. Wie in Fig. 3 gezeigt kann der Datensatz, der gerade analysiert wird, allgemein aus Eingangsdaten 44, die von dem allgemeinen Analyseprogramm 40 empfangen werden, wenn es zur Ausführung einer bestimmten Analyse (oder einer Reihe von Analysen) aufgerufen wird, und Parametern 46, auf die von dem allgemeinen Analyseprogramm 40 im Verlauf einer Ausführung von einer oder mehreren Analysen zugegriffen werden können, bestehen.

Die Eingangsdaten 44 können verschiedene Typen von Information in Abhängigkeit von der Wahl des Entwerfers und den Anforderungen einer bestimmten Anwendung enthalten. Allgemein können die Eingangsdaten 44 einen oder mehrere Datensätze (tatsächliche Parameter), die analysiert werden sollen, Zeiger oder Referenzen auf eine oder mehrere relevante Strukturen, die zu analysierende Datensätze enthalten, und irgendeine andere nützliche Information enthalten. Die Typen von Parametern 46 sind größtenteils für jede Anwendung des allgemeinen Analysesystems spezifisch. Wie in Fig. 4 gezeigt, können die Eingangsdaten 44 von der Einheit, z.B. einem

Computerprogramm, zugeführt werden, das das allgemeine Analyseprogramm 40 zum Ausführen einer Analyse aufruft, während die Parameter 46 von einer Anzahl von verschiedenen Quellen in der Anwendung zugeführt werden können.

Bezugnehmend als nächstes auf Fig. 4 lässt sich ein erweitertes schematisches Blockschaltbild des in einer vereinfachten Form in Fig. 3 gezeigten allgemeinen Analysesystems ansehen. Dieses Diagramm illustriert die Anwendung der vorliegenden Erfindung auf die Verarbeitung von Telefonanrufen in einer Telefonvermittlungsstelle oder einer Vermittlungsstelle. In Fig. 4 werden die Eingangsdaten 44 an das allgemeine Analyseprogramm 40 von einem Anrufsteuerprogramm 56 zugeteilt, welches eine Gesamtsteuerung der Verarbeitung von Telefonanrufen der Vermittlungsstelle übernimmt, einschließlich der Aufforderung einer Ausführung von Anrufen an dem allgemeinen Analyseprogramm 40, um verschiedene Analysen, z.B. eine Ziffern- und Wegleitungsanalyse (Routing), von jedem Telefonanruf auszuführen. In diesem Fall umfassen die Parameter 46 verschiedene Anruf-bezogene Daten, die in einer Anrufsteuerdatenstruktur 54 gespeichert werden. Wie in Fig. 4 gezeigt, können die in den Anrufsteuerdaten 54 gespeicherten Parameter 46 von dem allgemeinen Analyseprogramm 40 und dem Anrufsteuerprogramm 56 gelesen werden (auf sie kann zugegriffen werden).

In einer bevorzugten Ausführungsform der vorliegenden Erfindung, die nachstehend ausführlich diskutiert wird, werden eine Vielzahl von Anrufaufzeichnungen in den Anrufsteuerdaten 54 gespeichert. Jede Anrufaufzeichnung enthält einen Datensatz, d.h. einen Satz von Anrufparameterwerten, für einen spezifischen Anruf durch die Vermittlungsstelle (sämtliche Anrufaufzeichnungen enthalten den gleichen Satz von Anrufparameter-Stellen, aber die Parameterwerte in jeder Anrufaufzeichnung sind für einen spezifischen Anruf spezifisch). In der bevorzugten Ausführungsform umfassen die

Eingangsdaten 44 einen Zeiger oder eine Referenz auf die bestimmte Anrufaufzeichnung, d.h. den gewünschten Satz von Anrufparameterwerten 46, für den Anruf, der gerade analysiert wird (in der bevorzugten Ausführungsform umfassen die Analysesteuerdaten 42 mehrere verschiedene Analysestrukturen und Unterstrukturen und die Eingangsdaten 44 umfassen auch eine Identifikation der bestimmten Analysestruktur, die von dem Analyseprogramm 40 ausgeführt werden soll). Wie ausführlich nachstehend erläutert wird, verwendet das allgemeine Analyseprogramm 40 eine Parameteridentifikation (ID) 74, um auf den Wert eines bestimmten Parameters 46 von einer der Stellen in der Anrufaufzeichnung, angezeigt von den Eingangsdaten 44, zuzugreifen (diesen zu lesen).

Unter Bezugnahme wiederum auf Fig. 4 lässt sich ersehen, dass die Anrufsteuerdaten 54 Teilnehmerdaten 50 aus einer Teilnehmerdatenbank 52 und einen Teilnehmereingang 58 von einem Telefonteilnehmer, z.B. die Ziffern oder Zahlen, die von einem einleitenden Teilnehmer (A-Teilnehmer) unter Verwendung der Telefoneinheit 10 eingegeben (gewählt) werden, umfassen. Die Teilnehmerdaten 50 können zum Beispiel den Ortscode des A-Teilnehmers, die Dienstklasse (Geschäfts- oder Privatkunde) und andere ähnliche Information umfassen. Jede Anrufaufzeichnung in den Anrufsteuerdaten 54 kann deshalb sowohl "vorübergehende" Daten, die nur für einen bestimmten Anruf gültig sind, d.h. eine Teilnehmereingabe 58, als auch "semi-permanente" Daten, die für sämtliche Anrufe von einem bestimmten Teilnehmer gültig sind, d.h. Teilnehmerdaten 50, speichern.

Zusätzlich zu den Teilnehmerdaten 50 und der Teilnehmereingabe 58 können die Anrufsteuerdaten 54 auch die Zwischenergebnisse oder Endergebnisse 48 einer vorhergehenden Analyse, die von dem allgemeinen Analyseprogramm unter der Steuerung der Analysesteuerdaten 42 ausgeführt wird, andere Anrufverarbeitungsergebnisse und andere Daten, die in die Anrufsteuerdaten 54 von dem Anrufsteuerprogramm 56 geschrieben

werden, umfassen. Die aus irgendeiner Analyse erzeugten Ergebnisse können von dem allgemeinen Analyseprogramm 40 in die Anrufsteuerdaten 54 geschrieben und später von dem allgemeinen Analyseprogramm 40, um eine nachfolgende Analyse auszuführen, oder von dem Anrufsteuerprogramm 56, um irgendeinen externen Prozess zu steuern, verwendet werden. Zum Beispiel kann das allgemeine Analyseprogramm 40 die Ergebnisse der Zielstellenanalyse in die Anrufsteuerdaten 54 zur Verwendung bei der Ausführung einer Wegleitungs-Analyse schreiben. Die Ergebnisse der Wegleitungs-Analyse können dann von dem allgemeinen Analyseprogramm 40 in die Anrufsteuerdaten 54 geschrieben werden und von dem Anrufsteuerprogramm 56 verwendet werden, um eine Bündelleitung zum Transportieren des Anrufs durch das Netz zu wählen.

Gemäss der vorliegenden Erfindung verwendet das allgemeine Analyseprogramm 40 die Eingangsdaten 44, um irgendeine einer großen Anzahl von Analysen, die unter der Steuerung der Analysesteuerdaten 42 ausgeführt werden sollen, anzuzeigen und ferner einen Satz von Parametern 46 in den Anrufsteuerdaten 54, auf denen die angezeigte Analyse ausgeführt werden soll, allgemein zu bezeichnen. Jedoch werden die spezifische Sequenz oder die Reihenfolge, in der die bestimmten Parameterwerte analysiert werden, und die spezifische Analyse, die für jeden Parameterwert durch das allgemeine Analyseprogramm 40 ausgeführt werden soll, beide von den Analysesteuerdaten 42 bestimmt. Für den Fall einer Analyse einer Ziffer (anrufende Nummer oder angerufene Nummer) werden zum Beispiel die Reihenfolge, in der die Ziffern analysiert werden sollen, und die für jede Stelle, die sich unter einer Analyse befindet, auszuführenden Vorgänge von den Analysesteuerdaten 42 oder genauer gesagt von bestimmten Analysemodulen angezeigt, die sich in der Analysesteuerdatenstruktur für eine Ziffernanalyse befinden.

Es ist dieses Merkmal der Analysesteuerdaten 42, das die Verwendung eines generischen Programms, d.h. des allgemeinen

Analyseprogramms 40 zum Ausführen von unterschiedlichen Typen von Analysen, und die Freiheit zum Hinzufügen von neuen Parametern und zum Erzeugen von neuen Typen von Ergebnissen mit nur einer Mensch-Maschine-Schnittstelle erlaubt. Der Betreiber muss nur das Format einer Datenstruktur, d.h. der Analysesteuerdaten 42, kennen und kann leicht existierende Funktionen modifizieren und neue Funktionen während eines tatsächlichen Betriebs unter Verwendung der Mensch-Maschine-Schnittstelle hinzufügen, ohne den Code für das allgemeine Analyseprogramm 40 zu modifizieren oder eine einzelne Codezeile zu diesem hinzuzufügen. In einer Hinsicht wirkt das allgemeine Analyseprogramm 40 lediglich als ein Interpretierer der Analysesteuerdaten 42. Eine Ausführungsform des allgemeinen Analyseprogramms 40, der Analysesteuerdaten 42 und der Anrufsteuerdaten 54 ist in den Fig. 5-6 dargestellt.

Unter Bezugnahme auf die Fig. 5 lässt sich eine bildliche Darstellung der Konstruktion des allgemeinen Analyseprogramms 40 und der Analysesteuerdaten 42, die in den Fig. 3-4 gezeigt sind, nun ansehen. Das allgemeine Analyseprogramm 40 kann auf einer Vielzahl von Unterprogrammen aufgebaut sein, die nachstehend als "Analyse-Primitive" oder "Analyse-Primitiv-Funktion" bezeichnet werden. Die Analysesteuerdaten 42 können aus einer Vielzahl von Elementen konstruiert sein, die hier als "Analysemodule" 70 bezeichnet werden. In der bevorzugten Ausführungsform der vorliegenden Erfindung werden mehrere Kategorien von Analyse-Primitiven 78 und Analysemodulen 70 verwendet. Insbesondere können die Analyse-Primitive 78 entweder "Analyse-Primitive auf der unteren Ebene (Low-Level)" oder "Analyse-Primitive auf der oberen Ebene (High-Level)" sein. Ferner können die Analysemodule 70 Analysemodule auf der oberen Ebene (High-Level) oder Analysemodule auf der Steuerebene (Control-Level) sein. Außer wenn der Zusammenhang dies anders erfordert, sollen die Ausdrücke "Analyse-Primitive" und "Analysemodule" nachstehend dafür verwendet werden, um auf irgendeine oder sämtliche Kategorien vom Analyse-Primitiven bzw. Analysemodulen zu verweisen, während

auf jede einzelne Kategorie mit einem geeigneten Modifizierer, d.h. "Low-Level (untere Ebene)", "High-Level (obere Ebene)" oder "Control-Level (Steuerebene)" Bezug genommen wird.

Jede der Kategorien von Analyse-Primitiven 78 und Analysemodulen 70 spielt eine wesentliche Rolle innerhalb der bevorzugten Ausführungsform der vorliegenden Erfindung, die nachstehend ausführlich beschrieben wird. In der bevorzugten Analyse sind die Low-Level und High-Level-Analyse-Primitive 78 eingebaute Primitiv-Funktionen mit unterschiedlichen Typen. Die Low-Level-Analyse-Primitive 78 können zum Bilden von Low-Level-Analysemodulen 70 verwendet werden, die wiederum zum Bilden von High-Level Analysemodulen 70 verwendet werden können. Die vom Benutzer gebildeten High-Level-Analysemodule 70 und die eingebauten High-Level-Analyse-Primitive 78 können beide zum Bilden von verschiedenen Typen von Steuerebenen-Analysemodulen 70 zum Steuern einer bestimmten Analyse verwendet werden. Für die Zwecke der vorliegenden Diskussion kann Fig. 5 betrachtet werden, um einen gemeinsamen Typ von Steuerebenen-Analysemodul 70 in den Analysesteuerdaten 42 zu zeigen.

Bezugnehmend wieder auf die Fig. 5 weist jedes der Steuerebenen-Analysemodule 70 eine vorgegebene Struktur auf, die einen bestimmten Teil der Gesamtanalyse steuert, z.B. das Analysieren einer Ziffer in einer Telefonnummer. Es lässt sich ersehen, dass das beispielhafte Steuerebenen-Analysemodul 70 in Fig. 70 einen Funktionsanzeiger 72 und ein Steuerfeld 60 mit variablen Inhalten umfaßt, die zum Beispiel eine Parameteridentifikation (ID) 74 und andere Steuerdaten 76 enthalten können. Es lässt sich auch ersehen, dass das Steuerebenen-Analysemodul 70 von einem Modulanzeiger 80 identifiziert wird, der in dem Steuerfeld 60 von wenigstens einem anderen Steuerebenen-Analysemodul 70 in den Analysesteuerdaten 42 enthalten sein kann.

Unter Fortsetzung der Bezugnahme auf Fig. 5 bezeichnet der Funktionsanzeiger 72 (verweist auf) eines der High-Level-Analyseprimitive 78, die in dem allgemeinen Analyseprogramm 40 vorliegen. Wie sich aus der Diskussion der bevorzugten Ausführungsform deutlich ergeben wird, kann jedoch der Funktionsanzeiger 72 anstelle davon von eines der High-Level-Analysemodule 70 bezeichnen, die in einer ähnlichen Weise wie die High-Level-Analyse-Primitive 78 arbeiten, die aber in den Analysesteuerdaten 42 angeordnet sind. Das High-Level-Analyse-Primitiv 78, auf das Bezug genommen wird, oder das High-Level-Analysemodul 70, auf das Bezug genommen wird, bestimmt die spezifische Art und das Layout der Inhalte des Steuerfelds in dem Steuerebenen-Analysemodul 70, auf das Bezug genommen wird, und ist im Gegensatz dazu in der Lage, diese Inhalte zu interpretieren, d.h. in diesem Fall die Parameter-ID 74 und die Steuerdaten 76. Die Parameter-ID 74 zeigt die Stelle des Anrufparameterwerts (einen Datenuntersatz) an, der von dem Steuerebenen-Analysemodul 70 analysiert werden soll. Die Steuerdaten 76 können zum Beispiel die Modulanzeiger 80 von Analysemodulen 70 umfassen, aus denen ein nächstes Analysemodul 70 für eine Ausführung gewählt wird, auf Grundlage der Analyse des Parameterwerts, der von der Parameter-ID 74 angezeigt wird.

Als nächstes wird auf Fig. 6 Bezug genommen, wo eine bildliche Darstellung der Konstruktion der zunächst in Fig. 4 gezeigten Anrufsteuerdaten 54 nun ersehen werden kann. Wie vorher im Zusammenhang mit Fig. 4 diskutiert, umfassen die Steuerdaten 54 Teilnehmerdaten 50, eine Teilnehmereingabe 58, frühere Analyseergebnisse, die von dem allgemeinen Analyseprogramm 40 geschrieben werden, andere Verarbeitungsergebnisse und andere Daten, die von dem Anrufsteuerprogramm 56 geschrieben werden. Somit können sämtliche Daten, die sich auf einen Anruf von einem Telefonteilnehmer beziehen, in der Anrufsteuerdatenstruktur 54 gespeichert werden und auf sie kann von dem allgemeinen Analyseprogramm 40, mit Hilfe einer Parameter ID 74 während der Analyse des Anrufs zugegriffen

werden. Wie in Fig. 6 gezeigt, umfassen die Anrufsteuerdaten 54 eine Anrufaufzeichnung 116 für jeden Anruf, der gerade analysiert wird. Die Anrufaufzeichnung 116 umfasst einen Satz von Parameterwerten, die jeweils von einer einzigartigen Parameter-ID 74 identifiziert und assoziiert werden, d.h. die Anrufaufzeichnung 116 kann als eine Tabelle von Parameter-IDs 74 und entsprechenden Anrufparameterwerten visualisiert werden.

Unter Fortsetzung der Bezugnahme auf Fig. 6 wird die Analyse jedes Anrufparameterwerts, d.h. von einem der Werte in der Anrufaufzeichnung 116, von wenigstens einem Steuerebenen-Analysemodul 70 in den Analysesteuerdaten 42 gesteuert. Das High-Level-Analyse-Primitiv 78 (oder das High-Level-Analysemodul 70), auf das von dem Funktionsanzeiger 72 in dem Steuerebenen-Analysemodul 70 verwiesen wird, kann den gewünschten Parameterwert aus der Anrufaufzeichnung 116 durch Vergleichen der in dem Steuerebenen-Analysemodul 70 gefundenen Parameter-ID 74 mit der entsprechenden Parameter-ID 74, die zu diesem Wert in der Anrufaufzeichnung 116 gehört, zurückgeholt (gelesen) werden. Sobald der Parameterwert zurückgeholt worden ist, kann er von dem Analyse-Primitiv 78 verwendet werden, das das Steuerfeld 60 in dem Steuerebenen-Analysemodul 70 interpretiert, um zum Beispiel einen geeigneten Auslass von dem Steuerebenen-Analysemodul 70 zu wählen, d.h. den Modulanzeiger 80 des nächsten Analysemoduls 70, welches interpretiert werden soll, zu wählen.

Beispielhafte Analyse

Zu Beginn einer Analyse empfängt das allgemeine Analyseprogramm 40 einen Zeiger auf eine Analysestruktur in den Analysesteuerdaten 42 und einen Zeiger auf eine Anrufaufzeichnung (Parameteraufzeichnung) 116, d.h. einen Satz von Parametern 46, in den Anrufsteuerdaten 54. Das allgemeine Analyseprogramm interpretiert die Inhalte eines bestimmten Analysemoduls 70, z.B. des Analysemoduls 70 in den Fig. 5-6,

in den Analysesteuerdaten 42. Der von der Parameter-ID 74 in dem Analysemodul 70 angezeigte Parameter wird aus den Anrufsteuerdaten 54 geholt und das Analyse-Primitiv 78, das von dem Funktionsanzeiger 72 angezeigt wird, wird aus dem allgemeinen Analyseprogramm 40 aufgerufen und ausgeführt. Die Analyse wird dann mit dem nächsten Analysemodell-Modul 70 fortgesetzt, das von dem Modulanzeiger 80 angezeigt wird, der aus dem Steuerfeld 60 des Steuerebenen-Analysemoduls 70 gewählt wird. Dieser Prozess wird in der gleichen Weise wiederholt, bis die Analyse abgeschlossen ist. Demzufolge kann irgendeine gewünschte Analyse konstruiert werden, indem eine Anzahl von Analysemodulen 70, eins nach dem anderen, über den Modulanzeiger (über die Modulanzeiger) 80 in ihren Steuerfeldern 60 verbunden werden. Mit anderen Worten, wenn ein oder mehrere Analysemodule 70 einen Auslass zu einem anderen Analysemodul 70 aufweist, d.h. einen Modulanzeiger 80 umfasst, der auf ein anderes Analysemodul 70 weist, kann eine Analyse ausgeführt werden, indem ein Pfad von Auslässen durch die Analysesteuerdaten auf Grundlage der Parameter 46, die von den Eingangsdaten 44 und anderen Faktoren angedeutet werden, wie in Fig. 7 gezeigt, gewählt wird.

Unter Bezugnahme als nächstes auf Fig. 7 kann eine bildliche Darstellung der Wechselwirkung zwischen dem allgemeinen Analyseprogramm 40, den Analysesteuerdaten 42 und den Anrufsteuerdaten 54 in einer beispielhaften Analyse, die von dem allgemeinen Analysesystem der vorliegenden Erfindung ausgeführt wird, nun ersehen werden. In Fig. 7 sind drei beispielhafte Analysemodule 70 in den Analysesteuerdaten 42 (nicht gezeigt) der Deutlichkeit und Zweckdienlichkeit wegen als Analysemodule 82, 84, 86 getrennt bezeichnet und zwei beispielhafte Analyse-Primitive 78 in dem allgemeinen Analyseprogramm 40 sind ebenfalls der Deutlichkeit und Zweckdienlichkeit wegen als Analyse-Primitive 88, 90 getrennt nummeriert. Zum Zwecke der Vereinfachung wird angenommen, dass das Analyse-Primitiv 88 von dem Typ "Wähler" ist, der einen Auslass, d.h. einen Modulanzeiger 80, aus dem Analysemodul 70,

auf das Bezug genommen wird, zu dem nächsten Analysemodul 70 auf Grundlage des Parameterwerts, der gerade von dem Analysemodul 70, auf das Bezug genommen wird, analysiert wird, wählt. Das Analyse-Primitiv 90 ist andererseits ein Ergebnis-erzeugendes Analyse-Primitiv, das den Wert eines Zwischenergebnisses oder Endergebnisses in die Anrufsteuerdaten 54 schreibt.

Zu Beginn der beispielhaften Analyse interpretiert das allgemeinen Analyseprogramm 40 ein Analysemodul 82, um die Funktion (Analyse-Primitive 78) zu identifizieren, die ausgeführt werden soll. Weil der Funktionsanzeiger 72 in dem Analysemodul 82 auf das Analyse-Primitiv 88 verweist, führt das allgemeine Analyseprogramm 40 das Analyse-Primitiv 88 aus, welches wiederum den Parameterwert verwendet, der von der Parameter ID 74 in dem Analysemodul 82 identifiziert wird, um ein Analysemodul 84 als das nächste Analysemodul 70 in der beispielhaften Analyse zu wählen. An diesem Punkt interpretiert das allgemeine Analyseprogramm 40 ein Analysemodul 84 und führt wiederum ein Analyse-Primitiv 88 aus, das von dem Funktionsanzeiger 72 in dem Analysemodul 84 angezeigt wird. Auf Grundlage des Parameterwerts, der von der Parameter-ID 74 in dem Analysemodul 84 identifiziert wird, wird das Analysemodul 86 als das nächste Analysemodul 70 einer beispielhaften Analyse gewählt. Die beispielhafte Analyse geht entlang dieser Zeilen weiter und ein Zwischenergebnis oder ein Endergebnis kann schliesslich erzeugt und in die Anrufsteuerdaten 54 von dem Analyse-Primitiv 90, das das Ergebnis erzeugt, geschrieben werden, wenn die Analyse zum Beispiel durch das Analysemodul 86 geht.

Unter Bezugnahme auf Fig. 8 als nächstes lässt sich eine bildliche Darstellung der Konstruktion einer Zielstellenanalyse aus einer Kette von Steuerebenen-Analysemodule der Form, die allgemein in den Fig. 5-7 gezeigt ist, nun ansehen. Fig. 8 zeigt eine Art der beispielhaften Analyse, die allgemein in Fig. 7 gezeigt ist. In Fig. 8 sind

nur vier beispielhafte Steuerebenen-Analysemodule 70 gezeigt und der Klarheit wegen getrennt als Analysemodule 92, 94, 96, 98 bezeichnet. Die Steuerebenen-Analysemodule 92 und 96 sind von einem Typ, während die Steuerebenen-Analysemodule 94 und 98 von einem anderen Typ sind. Insbesondere verweist der Funktionsanzeiger 72 in jedem der Steuerebenen-Analysemodule 92 und 96 auf einen Typ von High-Level-Analyse-Primitiven 78, die als "Wähler" bezeichnet sind, während der Funktionsanzeiger 72 in jedem der Steuerebenen-Analysemodule 94 und 98 auf einen anderen Typ von High-Level-Analyse-Primitive 78, die als "Endergebnis" bezeichnet werden, verweist (sowohl der Wähler als auch das Endergebnis könnten jedoch an Stelle von High-Level-Analyse-Primitiven 78 High-Level-Analysemodule 70 sein). Die Funktion des Wähler-Analyseprimitivs 78 besteht darin, einen der Modulanzeiger 80 aus dem Steuerfeld 60 des Steuerebenen-Analysemoduls 70, auf das Bezug genommen wird, auf Grundlage des Wert des Parameters, der von der Parameter-ID 74 in dem Steuerebenen-Analysemodul 70, auf das Bezug genommen wird, zu wählen. Die Funktion des Endergebnis-Analyseprimitivs 78 besteht andererseits darin, einen vorgegebenen Wert von dem Steuerfeld 60 des Steuerebenen-Analysemoduls 70, auf das Bezug genommen wird, dem Parameter zuzuweisen, der von der Parameter-ID 74 in dem Steuerebenen-Analysemodul 70, auf das Bezug genommen wird, angezeigt wird.

Wie voranstehend erwähnt, hängt das Format des Steuerfelds 60 in irgendeinem Steuerebenen-Analysemodul 70 von dem Typ des High-Level-Analyse-Primitivs 78 (oder dem High-Level-Analysemodul 70), auf das von dem Funktionsanzeiger 72 Bezug genommen wird, ab. Im Fall der Wähler-Typ-Steuerebenen-Analysemodule 92 und 96 umfasst das Steuerfeld 60 eine Parameter-ID 74 und eine Vielzahl von Modulanzeigern 80, die auf mehrere andere Steuerebenen-Analysemodule 70 verweisen, die Kandidaten sind, um das nächste Analysemodul 70 in der Analysekette zu sein. In Abhängigkeit von dem Wert des Parameters, der von der Parameter-ID 74 in jedem der

Steuerebenen-Analysemodule 92 und 96 angezeigt wird, kann als das nächste Analysemodul 70 in der Analysekette jedesmal dann, wenn das Steuerebenen-Analysemodul 92 bzw. 96 ausgeführt wird, ein anderes Analysemodul 70 gewählt werden.

Das Steuerfeld 60 in jedem der Endergebnis-Typ Steuerebenen-Analysemodule 94 und 98 enthält andererseits eine Parameter-ID 74 und einen vorgegebenen Wert, der in die Stelle für den Parameter, der von der Parameter-ID 74 angedeutet wird, geschrieben werden wird, wenn das entsprechende Steuerebenen-Analysemodul 94 oder 96 ausgeführt wird. Die Analysemodule 94 und 98 in Fig. 8, die Ergebnisse erzeugen, werden verwendet, um Endergebnisse zu erzeugen, d.h. ihre Funktionsanzeiger 72 verweisen auf ein Endergebnis-Typ Analyseprimitiv 78 und bilden somit Endpunkte in der Analyse. Beispiele von Analyse-Primitiven 78, die Zwischenergebnisse erzeugen, können dem hier beiliegenden Anhang A entnommen werden. Die Analysemodule, die zum Erzeugen von Zwischenergebnissen verwendet werden, sind im Aufbau ähnlich zu den Analysemodulen 94 und 98, außer, dass sie einen Modulanzeiger auf das nächste Analysemodul 70 benötigen, da sie keine Endpunkte in der Analyse sind.

In Fig. 8 werden die Steuerebenen-Analysemodule 92, 94, 96, 98 zum Ausführen einer Zielanalyse eines Anrufs, z.B. der Ziffern einer Telefonnummer, die von dem A-Teilnehmer unter Verwendung der Telefoneinheit 10 gewählt werden, verwendet. Einige der relevanten Parameter in der Anrufaufzeichnung 116 für den Anruf, der gerade analysiert wird, sind in Fig. 8 gezeigt. Für die Zwecke einer beispielhaften Zielstellenanalyse in Fig. 8 sind nur vier Parameter so dargestellt, dass sie in der Anrufaufzeichnung 116 gespeichert sind, obwohl in der Praxis viele andere Parameter in der Anrufaufzeichnung 116 gespeichert werden können. Den vier beispielhaften Anrufparametern sind die externen Namen "Zielstelle", "Ziffer 1", "Ziffer 2" und "Ziffer 3" gegeben, die in entsprechende Parameter-Ids 74 (Zeiger) zusammengestellt (kompiliert) sind.

Der Parameter "Zielstelle" bezieht sich auf das Endergebnis 48 der Zielstellenanalyse. Die Stelle dieses Parameters in der Anrufaufzeichnung 116 empfängt ihren Wert an dem Ende einer Zielstellenanalyse. Die Parameter "Ziffer 1", "Ziffer 2" und "Ziffer 3" beziehen sich jeweils auf die ersten, zweiten und dritten gewählten Ziffern in der Teilnehmereingabe 58. Die Stellen dieser Parameter in der Anrufaufzeichnung 116 empfangen ihre Werte am Beginn des Anrufs.

Unter Bezugnahme auf Fig. 8 steuert das Analysemodul 92 die Analyse der ersten gewählten Ziffer, die mit dem Namen "Ziffer 1" identifiziert wird und dann gibt es insgesamt 10 mögliche Auslässe von dem Analysemodul 92 entsprechend zu den 10 möglichen ersten Ziffernwerten (0-9). Eine Ausführung des Analysemoduls 92 führt zu der Ausführung des Wähler-Analyse-Primitivs 78, das von dem Funktionsanzeiger 72 in dem Analysemodul 92 angezeigt wird. Wenn zum Beispiel die Ziffer 1 Null (0) ist, was einen Fernanruf anzeigt, dann wird das Analysemodul 94 als das nächste Analysemodul 70 in der Zielstellenanalyse gewählt. Das Analysemodul 94 enthält die Parameter-ID 74 "Zielstelle", die die Stelle in der Anrufaufzeichnung 116 identifiziert, wo das Ergebnis "Fernanruf" geschrieben werden sollte. Die Ausführung des Analysemoduls 94 führt zu der Ausführung des Endergebnis-Analyse-Primitivs 78, das den Wert "Fernanruf" dem Parameter "Zielstelle" zuweist (der Name "Fernanruf" wird in irgendeinen konstanten Wert zusammengestellt, der in die Stelle entsprechend zu der Parameter-ID "Zielstelle" geschrieben wird). In diesem Fall muss die Vermittlungsstelle keinerlei Zielstellenanalyse der zweiten, dritten etc. gewählten Ziffern ausführen, sondern einfach den Anruf mit der geeigneten Fernvermittlungsstelle in der Netzhierarchie verbinden.

Wenn andererseits die erste gewählte Ziffer Eins (1) ist, dann wird das Analysemodul 96 als das nächste Analysemodul 70 in der Zielstellenanalyse gewählt. Das Analysemodul 96 ist von dem gleichen Typ wie das Analysemodul 92, steuert aber die

Analyse der zweiten gewählten Ziffer, die mit dem Namen "Ziffer 2" identifiziert wird. Eine Ausführung des Analysemoduls 96 führt in ähnlicher Weise zu der Ausführung des Wähler-Analyseprimitivs 78, die von dem Funktionsanzeiger 72 in dem Analysemodul 96 angezeigt wird. Wenn die zweite gewählte Stelle ebenfalls 1 ist, was einen Anruf zu einem B-Teilnehmer anzeigt, der von einer anderen Vermittlungsstelle in dem gleichen Verzeichnisbereich, der von dem Nummerierungsplan zugewiesen wird, anzeigt, wird das Analysemodul 98 als das nächste Analysemodul 70 in der Zielstellenanalyse gewählt werden. Das Analysemodul 98 ist von dem gleichen Typ wie das Analysemodul 94, enthält aber das Ergebnis "andere Vermittlungsstelle" (anstelle "Fernanruf"), für die Information, die an die Stelle entsprechend zu der Parameter-ID 74 "Zielstelle" in der Anrufaufzeichnung 116 geschrieben werden sollte. Eine Ausführung des Analysemoduls 98 führt zu der Ausführung des Endergebnis-Analyseprimitivs 78, das den Wert "andere Vermittlungsstelle" dem Parameter "Zielstelle" zuweist. In diesem Fall muss die Vermittlungsstelle keinerlei weitere Zielstellenanalyse der dritten, vierten etc. gewählten Ziffern ausführen, sondern einfach den Anruf mit der anderen Vermittlungsstelle verbinden, die den angerufenen B-Teilnehmer bedient.

Fig. 8 zeigt das Fortschreiten der Zielstellenanalyse für nur zwei Fälle, d.h. wo die erste gewählte Stelle 0 ist oder wo die ersten und zweiten gewählten Stellen beide 1 sind. Wenn eine "Ziffer 1" den Wert Null aufweist, dann wird der Parameter "Zielstelle", nachdem die Zielstellenanalyse ihren Ablauf durchlaufen hat, den Wert "Fernanruf" aufweisen. Wenn andererseits sowohl die "Ziffer 1" als auch die "Ziffer 2" den Wert 1 aufweisen, dann wird der Wert des "Zielstellen"-Anrufparameters nach der Ausführung der Zielstellenanalyse eine "andere Vermittlungsstelle" sein. Der Einfachheit halber ist die Analyse von anderen Werten der "Ziffer 1" und "Ziffer 2" (oder im Grunde genommen die Analyse der übrigen Ziffern, d.h. "Ziffer 3") in Fig. 8 nicht gezeigt, es sei jedoch darauf hingewiesen, dass sie entlang ähnlicher Zeilen fortschreitet.

Wenn zum Beispiel die "Ziffer 1" weder 0 noch 1 ist, was einen Anruf zu einem B-Teilnehmer anzeigt, der von dieser Vermittlungsstelle bedient wird, würde einer der Modulanzeiger 80, der einem der Ziffernwerte 2-9 entspricht, gewählt werden und das angezeigte nächste Analysemodul 70 wird ausgeführt werden. Die Zielstellenanalyse wird fortgesetzt werden, bis eine oder mehrere der übrigen gewählten Ziffern analysiert worden ist und ein Endergebnis 48 in die Parameterstelle "Zielstelle" geschrieben worden ist. Die Vermittlungsstelle wird schließlich den Anruf direkt mit dem angerufenen Teilnehmer oder vielleicht mit einer PBX verbinden, mit dem der anrufende Teilnehmer verbunden ist.

Es lässt sich somit ersehen, dass die von dem allgemeinen Analysesystem der vorliegenden Erfindung ausgeführte Analyse als eine Baumstruktur dargestellt werden kann, die aus einer Vielzahl von Analysemodulen 70 (Knoten), die untereinander durch Zeiger (Verzweigungen), die aus ihren Steuerfeldern 60 abgeleitet sind, besteht. Die Inhalte eines bestimmten Steuerfelds 60 oder die Vorgehensweise, mit der der nächste Modulanzeiger 80 (wenn irgendeiner) davon abgeleitet wird, hängt von dem Typ des High-Level-Analyse-Primitivs 78 ab, das von dem Funktionsanzeiger 72 angezeigt wird. Für die in den Fig. 5-8 gezeigten Steuerebenen-Analysemodule 70 umfasst das Steuerfeld 60 eine einzelne Parameter-ID 74. Andere Steuerebenen-Analysemodule 70 können jedoch mehrere Anrufparameterwerte verwenden, d.h. mehrere Parameter-Ids 74 enthalten oder können alternativ irgendwelche Anrufparameterwerte überhaupt nicht verwenden. Mit anderen Worten, während jedes Steuerebenen-Analysemodul 70 in den Analysesteuerdaten 42 einen Funktionsanzeiger 72 enthält, ist die Parameter-ID 74 ein optionales Merkmal. Tatsächlich sind die Inhalte des gesamten Steuerfelds 60 in der Art optional und hängen von dem Typ von High-Level-Analyse-Primitiv 78 ab, das zum Interpretieren davon verwendet wird. Anders ausgedrückt, der Code für jeden Typ von High-Level-Analyse-

Primitiv 78 interpretiert eine gewisse entsprechende Struktur der Steuerebenen-Analysemodule 70.

Typen von High-Level-Analyse-Primitiven

Die High-Level-Analyse-Primitive 78, die als grundlegende Elemente bei verschiedenen Analysen verwendet werden können, d.h. zum Interpretieren von Steuerebenen-Analysemodulen 70 aufgerufen werden können, können von mehreren unterschiedlichen Typen sein, von denen einige eine ähnliche Funktion ausführen, d.h. eine Auswahl, eine Ergebniserzeugung etc. Die High-Level-Analyse-Primitive 78 können "Wähler" sein, die einen (oder mehrere) der Werte der Parameter 46 verwenden, um einen von mehreren möglichen Auslässen zu wählen, d.h. einen Modulanzeiger 80, in einem Steuerebenen-Analysemodul 70, und um dadurch die Analyse an das nächste Analysemodul 70, das von dem gewählten Modulanzeiger 60 angezeigt wird, zu richten. Ein Typ des Zählers kann einen der Parameter 46 als einen Index auf eine Tabelle von Auslässen in dem Steuerfeld 60 eines Analysemoduls 70 verwenden (zum Beispiel das "Wähler" Analyseprimitiv 78, welches im Zusammenhang mit Fig. 8 diskutiert wurde), während ein anderer Typ von Wähler den Wert des Parameters 46 mit einer Anzahl von Werten in dem Steuerfeld 60 vergleichen oder anpassen kann und einen Auslass auf Grundlage dieses Vergleichs wählen kann, der ein zeichenartiger, Bool'scher, bitartiger oder arithmetischer Vergleich sein kann. Andere Wähler können Verteiler sein, die zwischen Auslässen, zum Beispiel gemäß einer zirkularen, stochastischen oder prozentweisen Verteilung, verteilen.

Anderen Typen von High-Level-Analyse-Primitiven 78 können die Aufgabe einer Ergebniserzeugung zugewiesen werden, d.h. der Lieferung von Ausgangsdaten von der Ausführung eines Analysemoduls 70 an die Anrufsteuerdaten 54 (zum Beispiel das Analyseprimitiv 78 "am Endergebnis", welches im Zusammenhang mit Fig. 8 diskutiert wurde) oder die Aufgabe einer Analysesteuerung, d.h. einer Steuerung der Analysesequenz

einschließlich zum Beispiel des Aufrufens einer Gruppe oder einer Untergruppe von Analysemodulen 70 und des Zurückkehrens, wenn die Analyse der Gruppe oder der Untergruppe abgeschlossen worden ist. Die High-Level-Analyse-Primitive 78 können ferner bestimmte Analyseprimitivfunktionen ausführen, die im Gegensatz zu den Analysesteuerprimitiven, z.B. Wählern, die Wahl eines Pfads oder eine Route durch die Analyse nicht beeinflussen, sondern an Stelle davon eine bestimmte Funktion ausführen, zum Beispiel die Registrierung von Statistiken, das Berichten von Ergebnissen, die Modifikation von Parameterwerten oder die Übertragung von Signalen an andere Vermittlungsstellen in dem Netz.

Viel der High-Level-Analyse-Primitivfunktionen 78, die voranstehend erwähnt sind, werden weiter in dem hier beigefügten Anhang A diskutiert. Eine Diskussion des Indexwähler-Analyseprimitivs 78 lässt sich insbesondere auch in der Beschreibung der bevorzugten Ausführungsform, die folgt, finden. Von Durchschnittsfachleuten wird erkannt, dass die Typen von High-Level-Analyseprimitiven 78, die hier diskutiert werden, lediglich beispielhaft und nicht für die zahlreichen möglichen Typen, die in der vorliegenden Erfindung verwendet werden können, erschöpfend sind. Aus der hier vorliegenden Beschreibung der bevorzugten Ausführungsform der vorliegenden Erfindung lässt sich erkennen, dass die im Anhang A beschriebenen Funktionen in High-Level-Analysemodulen 70 implementiert werden können, die sich anstelle in den High-Level-Analyseprimitiven 78, die sich in dem allgemeinen Analyseprogramm 40 befinden, in den Analysesteuerdaten 42 befinden.

Typen von Steuerebenen-Analysemodulen

Wie die High-Level-Analyseprimitive 78 (oder die High-Level-Analysemodule 70), die diese interpretieren, kann es sich bei den Steuerebenen-Analysemodulen 70 um mehrere verschiedene Typen handeln. Für den Fall eines Steuerebenen-Analyse-Moduls

70, dessen Funktionsanzeiger 72 auf das Indexwähler-Analyseprimitiv 78 verweist, können die Steuerdaten 76 z.B. als eine Tabelle von Modulanzeigern 80 angeordnet werden und der Indexwähler 78 kann den Wert eines Parameters 6 verwenden, um die Tabelle zu indizieren und den entsprechenden Modulanzeiger 80 als einen Zeiger auf das nächste Analysemodul 70, welches ausgeführt werden soll (siehe Fig. 13), zu wählen. Wie in den Fig. 9A-9B gezeigt, kann jedoch das Steuerfeld 80 andere nützliche Informationen zusätzlich zu den Modul-Anzeigern 80 umfassen.

Unter Bezugnahme als nächstes auf die Fig. 9A-9B lassen sich nun schematische Darstellungen von zwei beispielhaften Verwendungen der Steuerdaten 76 in dem Steuerfeld 60 eines Analysemoduls ansehen. Fig. 9A veranschaulicht die Anordnung der Inhalte des Steuerfelds 60 eines Analysemoduls 70 für den Fall, dass der Funktionsanzeiger 72 auf ein Wähler-Analyseprimitiv 78 für eine zirkulare Verteilung verweist, das im Gegensatz zu dem Indexwähler 78 nicht einen Parameter 46 verwendet, um einen einer Vielzahl von Modulanzeigern 80 zu wählen, sondern anstelle davon zwischen den Modulanzeigern 80 in dem Steuerfeld 60 zyklisch wählt. Das Steuerfeld 60, das in Fig. 9A gezeigt ist, umfasst eine Vielzahl von Unterfeldern oder Schlitten 76a-f, die andere Steuerdaten zusätzlich zu den Anzeigern 80 für das nächste Modul speichern. In dem Beispiel für die zirkulare Verteilung, die in Fig. 9A gezeigt ist, enthält ein Schlitz 76a eine Bezeichnung der Anzahl von möglichen Auslässen, d.h. den nächsten Analysemodulen 70 und ein Schlitz 76b enthält eine Referenz auf den zuletzt verwendeten Auslass von dem Analysemodul 70, d.h. des nächsten Analysemoduls 70, das beim letzten mal gewählt wurde, als der Analysepfad das in Fig. 9A gezeigte Analysemodul 70 durchquerte, während jeder der Schlitze 76c-f einen Modulanzeiger 80 enthält, der auf einen der möglichen Auslässe von dem Analysemodul 70 verweist. In diesem spezifischen Beispiel gibt es vier mögliche Auslässe und der zuletzt

verwendete Auslass war zu dem Analysemodul 70, das von dem Modulanzeiger 80 in Schlitz 76d angezeigt wird.

Wenn die Analyse durch das in Fig. 9A gezeigte Analysemodul 70 geht, wird der zirkulare Verteiler 78 den Modulanzeiger 80 wählen, der sequentiell dem letzten verwendeten Modulanzeiger 80 in der zirkularen Verteilung folgt, und wird die Referenz auf den zuletzt verwendeten Auslass im Schlitz 78b modifizieren, um auf den gewählten Modulanzeiger 80 zu verweisen. In einer Implementierung kann die Referenz im Schlitz 76b einfach der Wert eines Zählers sein, der jedes Mal dann weitergestuft oder inkrementiert wird, wenn die Analyse durch das in Fig. 9A gezeigte Analysemodul 70 geht. Der im Schlitz 76a gespeicherte Wert wird dann eine obere Grenze für den Wert des Zählers darstellen, der auf eine untere Grenze immer dann zurückgesetzt wird, wenn diese obere Grenze durchlaufen worden ist, um so den Zählwert zyklisch zu durchlaufen. In dem in Fig. 9A gezeigten Beispiel wird der zirkulare Verteiler 78 den Modulverteiler 80 im Schlitz 76e wählen und die Referenz im Schlitz 76b aktualisieren, um auf den Schlitz 76e zu verweisen. Wie ersichtlich ist, wird die Auswahl des Auslasses, d.h. eines nächsten Analysemoduls 70 von dem zirkularen Verteiler 78 nicht unter Bezugnahme auf einen Parameter 46 durchgeführt, sondern ist von dem Wert des Zählers in dem zuletzt verwendeten Auslassschlitz 76b abhängig.

Fig. 9B zeigt eine andere Verwendung der Steuerdaten in dem Steuerfeld 60. Insbesondere veranschaulicht die Fig. 9B die Anordnung der Inhalte des Steuerfelds für den Fall, dass der Funktionsanzeiger 72 auf ein zeitabhängiges Wähler-Analyseprimitiv 78 verweist, das einen Wähleranzeiger 80 in dem Steuerfeld 60 auf Grundlage der Tageszeit in dem Moment wählt, wenn die Analyse durch das Analysemodul 70 geht, das in Fig. 9B gezeigt ist. In dieser Weise kann ein bestimmtes Wegleitungs-Muster (Routing-Muster) für Anrufe, die während eines bestimmten Teil des Tages verarbeitet werden, und ein anderes Wegleitungs-Muster, das für Anrufe gewählt wird, die

während eines anderen Teils des Tages verarbeitet werden, gewählt werden. Z.B. können Anrufe, die von einem Telefonkunden während normaler Geschäftszeiten durchgeführt wird, eine normale Anrufbehandlung erfahren, während Anrufe, die nach dem Ende der Geschäftszeit durchgeführt werden, anstelle davon automatisch an ein Sprachpost-System oder an einen Beantwortungsdienst geleitet werden können. Das beispielhafte Analysemodul 70, das in Fig. 9B gezeigt wird, kann verwendet werden, um die gewünschte Zeitdifferenzierung in der Wegleitungs-Analyse, die während einer Anrufverarbeitung ausgeführt wird, zu erzielen.

Unter fortgesetzter Bezugnahme auf Fig. 9B umfasst das Steuerfeld 60 ein Unterfeld 76w, das die Anzahl von Alternativen oder Pfaden bezeichnet, die von dem zeitabhängigen Wähler 78 gewählt werden können. Zur Veranschaulichung sind 3 Alternativen X, Y und Z in Fig. 9B gezeigt. Jede Alternative besteht aus 3 Schlitzen, die einen Startzeitdatenwert, einen Endzeitdatenwert und eine Referenz auf das nächste Analysemodul 70, d.h. einen Modulanzeiger 80, zugeordnet der jeweiligen Alternative, enthalten. Die Startzeit- und Endzeitdaten definieren das Zeitintervall, in dem der zeitabhängige Wähler 78 das nächste Analysemodul wählen wird, das der Alternative zugewiesen ist. Zum Beispiel definiert der Start-Zeitdatenwert im Schlitz 76x-1 und der Endzeit-Datenwert im Schlitz 76x-2 das Zeitintervall, in dem der Modulanzeiger 80 im Schlitz 76x-3 gewählt werden wird. In ähnlicher Weise definiert der Startzeit-Datenwert im Schlitz 76y-1 und der Endzeit-Datenwert im Schlitz 76y-2 das Zeitintervall, in dem der Modulanzeiger 80 im Schlitz 76y-3 gewählt werden wird, und der Startzeit-Datenwert in Schlitz 76z-1 und der Endzeit-Datenwert im Schlitz 76z-2 definieren das Zeitintervall, in dem der Modulanzeiger 80 im Schlitz 76z-3 gewählt werden wird. Wenn somit die Analyse durch das Analysemodul 70 geht, das in Fig. 9B gezeigt ist, wird der zeitabhängige Wähler 78 die Tageszeit mit den Zeitintervallen vergleichen, die in den Alternativen X, Y und Z definiert

sind, und wird den Pfad wählen, der von dem Modulanzeiger 80 angezeigt wird, der zu der Alternative gehört, in dessen Zeitintervall die Tageszeit gerade fällt.

Die Fig. 9A-9B zeigen deshalb zwei Beispiele der Steuerebenen-Analysemodule 70, die keine Parameter IDs 74 enthalten, die auf bestimmte Typen von High-Level-Analyseprimitive 78 (oder äquivalent High-Level-Analyse-Module 70) verweisen, die überhaupt keine Parameter beim Interpretieren der bestimmten Steuerdaten 76 in den Steuerebenen-Analysemodulen 70, auf die Bezug genommen wird, verwenden. In dieser Hinsicht können die Steuerebenen-Analysemodule 70, die in den Fig. 9A-9B gezeigt sind, als vollständig "autark" angesehen werden, d.h. sie enthalten sämtliche Information, die von den High-Level-Analyse-Primitiven 78, auf die Bezug genommen wird (oder den High-Level-Analysemodulen 70, auf die Bezug genommen wird) benötigt wird. Die Steuerebenen-Analysemodule 70 in den Fig. 5-8 verweisen andererseits auf andere Typen von High-Level-Analyseprimitiven 78 (oder High-Level-Analysemodulen 70), die individuell die Verwendung eines einzelnen Parameterwert benötigen. Einige Beispiele von Steuerebenen-Analysemodulen 70, die individuell mehrere Anrufparameterwerte verwenden und deshalb mehrere Parameter-IDs 74 benötigen, können aus der Diskussion, im Anhang A dazu, der bestimmten High-Level-Analyseprimitivfunktionen 78 (oder äquivalent High-Level-Analysemodulen 70), die diese interpretieren, gewonnen werden. Diese Beispiele sind natürlich nur für Illustrationszwecke vorgesehen und umfassen den gesamten Bereich von möglichen Steuerebenen-Analysemodulen 70, die in der vorliegenden Erfindung verwendet werden können, nicht.

Typen von Low-Level-Analyseprimitiven

Die Analyseprimitive 78 der unteren Ebene (low level) bestehen allgemein aus Primitivoperationen, z.B. Addieren, Holen etc., die als grundlegende Bildungsblöcke in ausgefeilteren Operationen verwendet werden. Im Gegensatz zu den High-Level-

Analyseprimitiven 78 werden die Low-Level-Analyseprimitive 78 nicht direkt verwendet, um eine Analyse zu konstruieren, sondern sie werden anstelle davon verwendet, um rekursiv neue Low-Level-Analysemodule 70 in den Analysesteuerdaten 42 zu konstruieren. Die neuen Low-Level-Analysemodule 70 sind im wesentlichen Permutationen oder Kombinationen der Low-Level-Analyseprimitiven 78, die in ähnlicher Weise verwendet werden können, um rekursiv in den Analysesteuerdaten 42 neue High-Level-Analysemodule 70 zu konstruieren, die in der Funktion äquivalent zu den High-Level-Analyseprimitiven 78 in dem allgemeinen Analyseprogramm 40 sind. Die neuen High-Level-Analysemodule 70, die aus den neuen Low-Level-Analyseprimitiven 70 konstruiert werden können, können wiederum zusammen mit den High-Level-Analyseprimitiven 78 verwendet werden, um neue Steuerebenen-Analysemodule 70 zum Steuern einer bestimmten Analyse zusammenzusetzen (zu bilden), z.B. eines Steuerebenen-Analysemoduls 70, das die dritte Ziffer in einer Telefonnummer bei der Routing-Analyse analysiert. Der Funktionsanzeiger 72 in jedem dieser drei Steuerebenen-Analysemodule 70 wird dann auf die High-Level-Analyseprimitivfunktion 78 oder das High-Level-Analysemodul 70 verweisen, von dem das Steuerebenen-Analysemodul 70 gebildet wird.

Im Anhang C hierzu wird ein Satz von Low-Level-Analyseprimitivfunktion 78 kurz beschrieben. Dieser Satz ist jedoch nicht erschöpfend, sondern ist dafür vorgesehen, den Typ von Low-Level-Analyse-Primitivfunktionen 78 beispielhaft darzustellen, die innerhalb der bevorzugten Ausführungsform der vorliegenden Erfindung verwendet werden können, d.h. für die Anwendung des allgemeinen Analysesystems für die Steuerung einer Anrufverarbeitung in einem Kommunikationssystem oder, genauer gesagt, einer Vermittlungsstelle in einem zentralen Vermittlungsbüro. Wie sich von Durchschnittsfachleuten ersehen lässt, können mehrere andere Low-Level-Analyse-Primitivfunktionen 78 in dem allgemeinen Analysesystem der vorliegenden Erfindung implementiert werden, unabhängig davon,

ob für die beispielhafte Telefonanwendung oder irgendeine andere Anwendung.

Typen von Analysen

Die Typen von Analysen, die von dem allgemeinen Analysesystem der vorliegenden Erfindung ausgeführt werden können, verändern sich mit den Anforderungen von verschiedenen Anwendungen. Die bevorzugte Ausführungsform der vorliegenden Erfindung, die nachstehend beschreiben ist, ist hauptsächlich auf eine Anrufverarbeitung in Telekommunikationssystemen (z.B. einer lokalen oder Fernvermittlungsstelle, einer Paketvermittlungsstelle etc.) anwendbar und führt deshalb derartige Analysen wie eine Ziffernanalyse und eine Routing-Analyse unter anderen verwandten Analysen aus. Im Anhang B hierzu ist eine vereinfachtes Analysebeispiel für eine lokale Vermittlung in einem hypothetischen Telefonnetz beschrieben. Dieses Beispiel verwendet viele der High-Level-Analyse-Primitivfunktionen 78 (oder äquivalent der High-Level-Analysemodule 70), die im Anhang A beschrieben und aus den Low-Level-Analyseprimitivfunktionen 78, die im Anhang C beschrieben sind, konstruiert werden können, während die Typen von Analysen (und die Typen von Analyseprimitiven 78 und Analysemodulen 70, die jede Analyse bilden), die im Anhang B beschrieben sind, irgendwie spezifisch für die Telefonieanwendung sind, lässt sich erkennen, dass die vorliegende Erfindung verwendet werden kann, um völlig andere Typen von Analysen in anderen Anwendungen auszuführen.

Die bevorzugte Ausführungsform

Nachdem der Gesamtaufbau und Operation des allgemeinen Analysesystems der vorliegenden Erfindung beschrieben worden ist, wird sich die übrige Diskussion auf eine bestimmte bevorzugte Ausführungsform konzentrieren, die nur eine der Abänderungen von möglichen Implementierungen, Veränderungen und Designalternativen darstellt. Während viele Techniken zum

Implementieren der Konzepte des allgemeinen Analysesystems der vorliegenden Erfindung sich einem Durchschnittsfachmann nach Lesen der voranstehenden Diskussion ergeben, wird die Technik für die gefädelte Interpretationssprache (Threaded Interpretative Language TIL) und insbesondere die Forth Programmiersprache nachstehend verwendet, um einige der Konzepte zu illustrieren, die der bevorzugten Implementierung des allgemeinen Analysesystems zugrunde liegen. Bevor jedoch auf die bevorzugte Ausführungsform Bezug genommen wird, wird eine kurze Beschreibung von Systemen mit einem gefädelten Code (threaded code) die Grundlage für einige der Konzepte legen, die nachstehend eingeführt werden. Weiterer Hintergrund für den Gegenstand von Threaded-Code-Systemen lassen sich aus Kogge, Peter M. "An Architectural Trail to Threaded-Code-Systems", Computer, März 1982, S. 22-32 ansehen. Eine zusätzliche Verdeutlichung lässt sich aus dem Lesen der Beschreibung der Low-Level-Analyseprimitive entnehmen, die im Anhang C hierzu aufgeführt sind.

Threaded Interpretative Sprachen

Computersprachen sind im allgemeinen dafür vorgesehen, um die Brücke zwischen den Primitivoperationen, für deren Ausführung der Computer entworfen ist, z.B. Addieren, Holen aus einem Speicher etc., und den Aufgaben, die der Benutzer erzielen muss, z.B. Invertieren einer Matrix, Suchen einer Datei, etc., zu schlagen. Weil verschiedene Benutzer bei verschiedenen Anwendungen verschiedene Anforderungen aufweisen, ist es im Grunde genommen für eine nicht-erweiterbare Sprache unmöglich, gleichzeitig sämtliche von diesen Anforderungen zu erfüllen. Ein Ansatz für dieses Dilemma kann darin bestehen, dass versucht wird, in die Sprache sämtliche möglicherweise nützlichen Operationen, Strukturen und Einrichtungen einzubauen. Dieser Ansatz ist jedoch mühsam und mit einem geschlossenen Ende. Gefädelte interpretative Sprachen (threaded interpretative languages) nehmen einen anderen Ansatz mit offenem Ende an. Deren Ansatz besteht darin, einige

wenige Techniken bereitzustellen, die dem Nutzer ermöglichen, die speziellen Operationen, die aus einer bestimmten Anwendung benötigt werden, schnell hinzuzufügen. Die Fähigkeit zum Hinzufügen von Spracheinrichtungen und Compiler-Strukturen wird als Erweiterungs-Fähigkeit bezeichnet. Diese Erweiterungsfähigkeit kann auf wenigstens 2 Ebenen zum Erhöhen der Leistung, d.h. der Fähigkeit zum Hinzufügen von Spracheinrichtungen unter Verwendung von existierenden Compilern und der Fähigkeit zum Erzeugen von neuen anwendungsspezifischen Compilern, die wiederum verwendet werden können, um nützlichere Spracheinrichtungen hinzuzufügen, auftreten; siehe z.B. Harres, Kim, "Forth Extensibility: Or How to Write a Compiler in Twenty-Five Words or less" Byte, August 1980, S. 164-169.

Allgemein gesagt wird ein Computerprogramm, das in einer Threaded Interpretative Language (gefädelten Interpretationssprache TIL), z.B. einer Forth oder Post-Script Sprache, geschrieben ist, im Computer als eine Liste von Adressen von Unterprogrammen dargestellt, wobei jedes von diesen aus Adressen von anderen Unterprogrammen besteht, usw. bis die primitiven Operationen der Sprache erreicht werden. Weil er außer das Lesen von Adressen und das Springen zu diesen wenig zu tun hat, kann ein Threaded-Language-Interpreter sehr schnell arbeiten. Verschiedene Typen von Adresseninterpretierern werden in Threaded-Code Systemen verwendet. In einem direkten Threaded-Code System wird eine Sequenz von Unterprogrammen in eine Liste von ihren Eintrittsadressen zusammengefädelt. Genauer gesagt können mit einer einfachen Liste von Adressen die eingenesteten Unterprogrammaufrufe auf der vorletzten untersten Unterprogramm-Einnestungsebene (wo jedes aufgerufene Unterprogramm wiederum nicht andere Unterprogramme aufruft) in einem Computerprogramm, das in einer herkömmlichen High-Level Programmiersprache, z.B. ADA oder PASCAL, geschrieben ist, dargestellt werden. Eine kleine Maschinensprachenroutine (die als Interpretierer bekannt ist) geht durch diese Liste, wobei

sie sequentiell indirekte Verzweigungen in jedem Schritt ausführt. Ein Interpretierer-Register verweist auf die nächste Adresse in der Liste von auszuführenden Prozeduren, d.h. verweist auf die Adresse des tatsächlichen Maschinencodes für die nächste Prozedur. Eine Änderung an dieser Technik fügt einen Grad von fehlender Richtungsfähigkeit zu dem Adresseninterpretierer hinzu. In einem indirekten Threaded-Code-System verweist das Interpretiererregister nicht direkt auf die Adresse der nächsten Prozedur, sondern verweist anstelle davon auf die Adresse eines Zeigers zu der nächsten Prozedur. Dennoch kann ein weiterer Grad von fehlender Richtungsfähigkeit zu dem Adresseninterpretierer hinzugefügt werden. In einem indirekten Token-Threaded-Code-System verweist das Interpretiererregister auf die Adresse eines Tokens, das als ein Index zu einer Tabelle von möglichen Zeigern zu der nächsten Prozedur dient.

Eine TIL Anwendung wird im Hinblick auf einen Satz von anwendungs-spezifischen Worten (oder Objekten) definiert, die in einem Wörterbuch in dem Til System vorhanden sind. Jedes "Wort" in dem TIL Wörterbuch bezeichnet eine ausführbare Prozedur (nicht einfach ein Teil oder ein Byte eines Speichers), der ein externer alphanumerischer Name gegeben wird. Der interne TIL Wörterbuch-Eintrag für jedes Wort besteht allgemein aus einem Namenfeld, das den zusammengestellten Namen des Worts enthält, einem Code-Feld, das eine Referenz auf die Befehle, die ausgeführt werden sollen, enthält, wenn das Wort ausgeführt wird, und einem Parameter-Feld, das die zusammengestellte Definition des Worts, z.B. eine Reihe von Referenzen auf andere Wörter in der Definition, enthält, besteht. Das TIL Wörterbuch enthält einen ersten Satz von eingebauten Wörtern, die für die meisten Benutzeranwendungen grundlegend sind, und einen zweiten Satz von eingebauten Wörtern, die dem Benutzer ermöglichen, neue anwendungs-spezifische Worte, die zu dem Wörterbuch hinzugefügt werden sollen, zu erzeugen. Sobald ein Wort in dem TIL Wörterbuch definiert ist, kann es zum Definieren von

vielen anderen anwendungs-spezifischen Worten verwendet werden. Das TIL Wörterbuch ist deshalb erweiterbar.

Der Prozess zum Eingeben von neuen Wörtern in das TIL Wörterbuch wird normalerweise als der Zusammenstellungs-Zustand (Kompilierungszustand bezeichnet). In dem Kompilierungs-Zustand) liest das TIL System Definitionen von neuen anwendungs-spezifischen Worten, die von irgendeiner externen Quelle eingegeben werden, z.B. einer Massenspeichereinrichtung oder einem Videoterminal. Die Definition von jedem neuen Wort besteht aus einem definierenden Wort, gefolgt von dem Namen des Words, welches definiert wird, und dann von dem Definitionskörper für das neue Wort, z.B. anderen Worten und Zahlen. Das definierende Wort ist ein eingebautes oder benutzer-definiertes (anwendungs-spezifisches) Wort, dass das TIL System veranlasst, in den Kompilierungszustand überzugehen, um den Definitionskörper zu kompilieren und einen Eintrag für das Wort, das gerade definiert wird, in dem TIL Wörterbuch hinzuzufügen (der Eintrag wird die kompilierte Definition enthalten). Einige Beispiele von eingebauten definierenden Wörtern in der Forth Sprache sind ":" (Doppelpunkt), KONSTANT und VARIABLE. Beispiele von anderen eingebauten Vierten Wörtern umfassen ";" (Semikolon), welches die Definition eines neuen Words definiert mit (:) beendet. Ein Beispiel eines definierenden Words in der bevorzugten Ausführungsform der vorliegenden Erfindung ist PROC. Andere Wörter, die in der bevorzugten Ausführungsform verwendet werden, umfassen ENDPROC, was die Definition eines neuen Words definiert von PROC beendet.

Die Definition eines definierenden Words ist unterschiedlich zu derjenigen von anderen Wörtern dahingehend, dass sie sowohl einen BUILD(Bildungs- oder Kompilierungs-Teil der ausgeführt wird, wenn Worte von dem definierenden Wort kompiliert werden, und einen DO(Ausführungs- oder Interpretierungs)-Teil, der ausgeführt wird, wenn die von dem definierenden Wort

komplilierten Wörter ausgeführt werden, umfasst. Die Funktion des BUILD-Teils (die in dem Kompilierungszustand ausgeführt wird) besteht darin, die Definition eines neuen Worts in das TIL Wörterbuch hinein zu kompilieren. Die Kompilierung bringt allgemein ein Schreiben des Namens eines neuen Worts in das Namensfeld und ein Schreiben einer Referenz an den DO-Teil in das Code-Feld des Wörterbucheintrags für das neue Wort mit sich. Der BUILD-Teil des definierenden Wortes wird auch den Definitionskörper des neuen Worts in das Parameterfeld des entsprechenden Wörterbucheintrags in solcher Weise kompilieren, dass der DO-Teil das Parameterfeld immer dann interpretieren kann, wenn das neue Wort ausgeführt wird.

Wenn z.B. ein neues Wort mit dem definierenden Wort (:) in der Forth Programmiersprache oder dem definierenden Wort PROC in der bevorzugten Implementierung des allgemeinen Analysesystems der vorliegenden Erfindung kompiliert wird, wird der Definitionskörper des neuen Worts allgemein aus einer Sequenz von eingebauten oder benutzer-definierten (anwendungs-spezifischen) Wörtern bestehen, die alle nacheinander in dem TIL Wörterbuch nachgeschlagen werden, und für jeden Wortnachschatz wird die Adresse des entsprechenden Wörterbucheintrags in das Parameterfeld des Wörterbucheintrags für das neue Wort, welches gerade kompiliert wird, geschrieben. Wenn das neue Wort ausgeführt wird, liest der Interpretierer in dem DO-Teil von (:) oder PROC einfach diese Adressen und führt die Wörter, auf die Bezug genommen wird, aus. Für Definitionen, die von dem definierenden Wort VARIABLE in der Forth Programmiersprache kompiliert werden, kann jedoch der Definitionskörper von jedem neuen Wort die Größe der Variable enthalten, die dem Parameterfeld des neuen Wörterbucheintrags von dem BUILD-Teil des VARIABLE definierenden Worts zugewiesen werden sollte. Wenn das neue Wort, das von VARIABLE definiert wird, ausgeführt wird, wird der DO-Teil des VARIABLE definierenden Worts ausgeführt und kann die Adresse der Variable zurückgeben. Somit sind diese definierenden Worte tatsächlich verschiedene Compiler, d.h.

die definierenden Wörter (:) und PROC kompilieren
 Prozedurdefinitionen (Operatoren), während das definierende
 Wort VARIABLE Datendefinitionen (Operanden) kompiliert.

Das TIL System behandelt benutzer-definierte Worte genauso wie
 vom System zugeführte (eingebaute) Worte, d.h. sämtliche
 benutzer-definierten Operatoren und Operanden können genauso
 wie vom System zugeführte Versionen verwendet werden.

Demzufolge unterstützt das TIL System wenigstens zwei Ebenen
 einer Erweiterungsfähigkeit. Auf der ersten Ebene verwendet
 der Benutzer standardgemäße definierende Wörter, um zu dem TIL
 Wörterbuch neue Wörter (Prozeduren) hinzuzufügen, die im
 Hinblick vom System zugeführte oder vom Benutzer definierte
 Wörter definiert werden. Der Einbau eines vorher kompilierten
 Worts in die Definition eines nachher kompilierten Worts kann
 als analog zu dem Einsetzen von Unterprogrammaufrufen
 angesehen werden. Die Prozedur, die von irgendeinem Wort
 bestimmt wird (ob vom Benutzer definiert oder eingebaut), in
 dem TIL Wörterbuch wird aufgerufen, indem einfach der Name des
 Worts in der Definition eines neuen Worts verwendet wird. Auf
 der zweiten Ebene einer Erweiterungsfähigkeit erzeugt der
 Benutzer neu definierende Wörter, die wiederum verwendet
 werden, um neue "Familien"-Typen von Wörtern zu dem TIL
 Wörterbuch hinzuzufügen (zu kompilieren). Auf dieser höheren
 Ebene verwendet der Benutzer jedes neue definierende Wort, um
 zu spezifizieren, wie ein Mitglied jeder entsprechenden
 Familie, d.h. ein neues Wort, das von dem neuen definierenden
 Wort kompiliert wird, kompiliert und ausgeführt werden soll.

Das Verhalten zur Kompilierungszeit und das Verhalten zur
 Ausführungszeit jedes neuen Mitglieds einer Familie von Worten
 sind gewöhnlicherweise z.B. in Forth, durch Verwendung der
 speziellen (Schlüssel) Worte <BUILDS bzw. DOES> in der
 Definition jedes neuen definierenden Worts spezifiziert (die
 Zeichen "<" und ">" sind Teil der Namen von diesen
 Schlüsselwörtern und zeigen an, dass <BUILDS vor DOES kommt>
 oder mit anderen Worten, das der BUILD-Teil vor dem DO-Teil

kommt). Das Verhalten zur Kompilierungszeit wird von dem Wort <BUILD und irgendwelchen Worten, die bis zu dem Wort DOES> folgen spezifiziert. Eine Ausführung des definierenden Worts bildet eine Wörterbuchdefinition für ein neues Familienmitglied. Die Inhalte dieser Definition werden von den Kompilierungszeitwörtern konstruiert. Das Verhalten zur Ausführungszeit wird von dem Wort DOES> und irgendwelchen Worten, die bis zum Ende der Definition folgen, spezifiziert. Eine Ausführung von irgendeinem Familienmitglied führt die Ausführungszeitwörter aus. Wenn ein neues Mitglied kompiliert ist und das neue definierende Wort ausgeführt wird, wird die <BIULDS Prozedur ausgeführt. Die <BUILDS Prozedur liest den Namen des neuen Mitglieds, d.h. nächste Wort nach dem neuen definierenden Wort in dem Eingangstextstrom und fügt ein neues Namensfeld und ein Code-Feld für eine neue Definition am Ende des Wörterbuchs hinzu. Das Namensfeld enthält den Namen des neuen Mitglieds und das Code-Feld enthält einen Zeiger auf die Ausführungszeitwörter in der Wörterbuchdefinition des neu definierenden Worts, das ausgeführt werden wird, wenn das neue Mitglied ausgeführt wird. Wenn das neue Mitglied ausgeführt wird, dann wird die DOES> Prozedur gefolgt von den Wörtern zwischen DOES> und dem Ende der neuen definierenden Wortdefinition ausgeführt. Die DOES> Prozedur gibt die Adresse des Parameterfelds innerhalb der Wörterbuchdefinition des neuen Mitglieds zurück.

Die Prinzipien von TIL, einschließlich der Verwendung der <BUILDS und DOES> Prozeduren in der Forth-Sprache können auf bevorzugte Implementierungen des allgemeinen Analysesystems angewendet werden. Zum Beispiel können einige der Low-Level-Analyseprimitivfunktionen 78 im Anhang C hierzu z.B. DUPLICATE, so angesehen werden, dass sie grob dem Standard (eingebauten) Wörtern in TIL entsprechen. Andere Low-Level-Analyseprimitivfunktionen 78, z.B. PROC, können so angesehen werden, dass sie grob den eingebauten definierenden Wörtern in TIL entsprechen. Die High-Level-Analyseprimitivfunktionen 78 im Anhang A dazu und die High-Level-Analysemodule 70 können so

betrachtet werden, dass sie grob den ausgefilterten eingebauten und vom Benutzer definierten definierenden Wörtern jeweils in TIL entsprechen. Die Steuerebenen-Analysemodule 70 können so angesehen werden, dass sie grob den Wörtern in TIL entsprechen, die aus diesen ausgefilterten definierenden Wörtern definiert werden (Fälle davon sind).

Allgemeines Analysesystem

Als nächstes wird auf die Figuren 10-16 Bezug genommen, wo eine Reihe von bildlichen Darstellungen gesehen werden können, die jeweils den Aufbau von einer oder mehreren Komponenten oder Elementen der bevorzugten Ausführungsform für das allgemeine Analysesystem der vorliegenden Erfindung darstellen. Fig. 10 umfasst eine Illustration der Struktur der bevorzugten Ausführungsform für das allgemeine Analyseprogramm 40, während Fig. 11 den Aufbau der bevorzugten Ausführungsform für die Analysedaten 42 darstellt. Fig. 12 zeigt den Aufbau eines Typs von Steuerebenen-Analysemodulen 70 in der bevorzugten Ausführungsform der in Fig. 11 gezeigten Analysesteuerdaten 42, während Fig. 13 die Verwendung eines beispielhaften Steuerebenen-Analysemoduls 70 des in Fig. 12 gezeigten Typs illustriert. Fig. 14 illustriert die Vorgehensweise, mit der ein Low-Level-Analyse-Modul 70 aus Low-Level-Analyse-Primitiven 78 konstruiert werden kann, während Fig. 15 zeigt, wie das vom Benutzer definierte Low-Level-Analyse-Modul 70, das in Fig. 14 gezeigt ist, verwendet werden kann, ein High-Level-Analyse-Modul 70 zu bilden. Fig. 16 zeigt, wie ein Steuerebenen-Analysemodul 70, ähnlich wie das in Fig. 12 gezeigte beispielhafte Analysemodul 70, mit der Bildungsprozedur des High-Level-Analyse-Moduls 70, das in Fig. 15 gezeigt ist, zusammengesetzt werden kann. Die in Fig. 10-16 gezeigte bevorzugte Ausführungsform basiert teilweise auf den Techniken, die von indirekten token-threaded Code-Systemen implementiert werden, auf die voranstehend Bezug genommen wurde.

Zunächst beziehend auf Fig. 10 umfasst das allgemeine Analyseprogramm 40 einen Block mit einer generischen Analysesoftware 100 und einer Anzahl von Analyseprimitiven 78, die in der bevorzugten Ausführungsform mit 120 unnummeriert sind, wobei die High-Level-Analyse-Primitive 120 jeweils eine Do-Prozedur 112 und eine Build-Prozedur 114 umfassen. Wie sich nach einer Betrachtung der Fig. 10-16 und der beiliegenden Diskussion entnehmen lässt, ermöglicht ein physikalisches oder konzeptionelles Trennen der Analyseprimitive 120 von dem Softwareblock 100 in dem allgemeinen Analyseprogramm 40, dass neue Analyseprimitive 120 leicht in nachfolgenden Versionen des allgemeinen Analyseprogramms 40 hinzugefügt werden können, ohne die generische Analysesoftware 100 zu modifizieren.

Für Zwecke einer Illustration sind nur 4 beispielhafte High-Level-Analyseprimitive 120 in Fig. 10 gezeigt und deren Funktionen sind alphabetisch mit "A-D" bezeichnet. Es lässt sich aus der vorangehenden Diskussion jedoch ersehen, dass die Analyseprimitive 120 sowohl Low-Level- als auch High-Level-Analyseprimitive umfassen und dass ferner die Anzahl von High-Level-Analyseprimitivfunktionen 120 allgemein viel größer als vier in der Praxis sein muss. Der Softwareblock 100 umfasst eine Analyse-Bildungsprozedur 102 (Analyse-Build-Prozedur), die mit einer Betreiberschnittstelle 200 kommuniziert, z.B. gemäss dem Open System Interconnection (OSI) Modell. Wie vollständig nachstehend diskutiert wird, insbesondere im Zusammenhang mit der Beschreibung des Kompilierungs Zustands der vorliegenden Erfindung, kann die Analysebildungsprozedur 102 zusammen mit der Bildungsprozedur 114 (Build-Prozedur) einer High-Level-Analyseprimitivfunktion 120 verwendet werden, um neue Steuerebenen-Analysemodule 70 zusammenzustellen (zu bilden), die in der bevorzugten Ausführungsform mit 130 unnummeriert sind. Wie ferner nachstehend diskutiert wird, kann die Build-Prozedur 114 eines benutzer-definierten High-Level-Analysemoduls 130 in ähnlicher Weise verwendet werden, um neue Steuerebenen-Analysemodule 130 zu bilden.

Unter weiterer Bezugnahme auf Fig. 10 umfasst der Softwareblock 100 eine Analyse-Interpretiererprozedur 108, die einen Baum von Analysemodulen 130 im Ansprechen auf einen Befehl interpretiert, um eine bestimmte Analyse auszuführen. Der Befehl zum Ausführen einer bestimmten Analyse kann von irgendeinem "Analysebenutzer" gesendet werden, d.h. irgendeinem Typ von Programm in der Vermittlungsstelle, die (Anrufe für die Ausführung einer) eine Analyse verwendet. Ein typisches Beispiel eines Analysebenutzers ist das Anrufsteuerprogramm 56, welches analogen oder digitalen Verkehr steuert, z.B. Telefonanrufe, durch die Vermittlungsstelle. Andere Analysebenutzer können Betriebswartungsprogramme verschiedener Arten sein. Für Zwecke der vorliegenden Erfindung wird jedoch angenommen, dass der Befehl zum Ausführen einer bestimmten Analyse von dem Anrufsteuerprogramm 56 gesendet wird.

Gemäss der bevorzugten Ausführungsform sendet das Anrufsteuerprogramm 56 an die Analyse-Interpretiererprozedur 108 Eingangsdaten 44, die aus (i) dem externen (symbolischen) Namen für eine auszuführende Analyse z.B. "Ziffernanalyse" oder "Routing-Analyse", (ii) einem Parameteraufzeichnungszeiger und, für bestimmte Analysen, (iii) andere Information, die kontext-abhängig (analyse-spezifisch) ist, bestehen. Der symbolische Name der Analyse wird verwendet, um auf die bestimmten Analysesteuerdaten für die auszuführende Analyse zuzugreifen. Der Parameter-Aufzeichnungszeiger wird verwendet, um in den Anrufsteuerdaten 54 auf den Block von Parameterwerten für den zu analysierenden Anruf (die Anrufaufzeichnung) zuzugreifen, von dem oder zu dem die High-Level-Analyseprimitive 120 (oder die High-Level-Analysemodule 130) dann irgendwelche Anrufparameterwerte, die von den Parameter-IDs 74 in den Steuerebenen-Analysemodulen 130 für die bestimmte Analyse angezeigt werden, erhalten oder, je nach Anforderung, schreiben können.

Unter weiterer Bezugnahme auf Fig. 10 übersetzt ein Analyse-Primitivfunktionswörterbuch 104 den externen (symbolischen) Namen einer Analyseprimitivfunktion 120, z.B. EXTERN A, der von einem Betreiber über eine Betreiberschnittstelle 200 eingegeben wird, in ein Primitivfunktions-Token, z.B. eine Nummer (347), die die spezifische Analyseprimitivfunktion 120 identifiziert, die EXTERN A genannt wird. Das Primitivfunktions-Token dient als ein Index in einer Analyseprimitivfunktions-Tokentabelle 106, die verwendet wird, um eine der Analyseprimitivfunktionen 120 zu lokalisieren, einschließlich einer der High-Level-Analyseprimitivfunktionen "A-D", z.B. einer Analyseprimitivfunktion "A". Die Analyseprimitivfunktions-Tokentabelle 106 übersetzt den Wert des Primitivfunktions-Tokens, z.B. 347, in einen Zeiger zu der Low-Level-Analyseprimitivfunktion 120 oder in zwei Zeiger zu der High-Level-Analyseprimitivfunktion 120, die diesem Tokenwert zugewiesen ist.

Die zwei Zeiger zu der High-Level-Analyseprimitivfunktion 120, z.B. die Analyseprimitivfunktion "A", besteht aus einem ersten Zeiger zu der Do-Prozedur 112 und einem zweiten Zeiger zu der Build-Prozedur 114 der High-Level-Analyseprimitivfunktion 120. Wie ausführlich nachstehend erläutert wird, wird die Do-Prozedur 112 einer High-Level-Analyseprimitivfunktion 120 ausgeführt, wenn diese Analyseprimitivfunktion 120 verwendet wird, um eine Analyse auszuführen, während die Bildungsprozedur 114 (Build-Prozedur) der gleichen Analyseprimitivfunktion 120 ausgeführt wird, wenn dies Analyseprimitivfunktion 120 verwendet wird, um eine neues Steuerebenen-Analysemodul 130 zu kompilieren (zusammenzustellen). In dieser Hinsicht entsprechen die Analyseprimitive 78, die in den Fig. 7-8 gezeigt und voranstehend in dem allgemeinen Übersichtsabschnitt der Beschreibung diskutiert wurden, genauer dem DO-Prozeduren 112 der Analyseprimitivfunktionen 120, die hier in Fig. 10 gezeigt sind.

Level-Analyse-Primitivfunktion 120 (oder einem High-Level-Analysemodul 130) arbeitet, zugegriffen werden, um Daten aus der Anrufsteuerdatenbank 54 während einer Analyse zu lesen (zu holen) und Daten darin zu schreiben (zu legen), und von dem Anrufsteuerprogramm 56, um Daten für andere Zwecke zu lesen und zu schreiben. Zum Beispiel kann die Do-Prozedur 112 einer Analyseprimitivfunktion 120, z.B. die Analyseprimitivfunktion "A", den Analyseausführungs-Eingangsdaten-Behandler 110 verwenden, um irgendwelche der Parameterwerte in dem Anrufsteuerdatenblock 116 zu erhalten. In diesem Fall wird die Do-Prozedur 112 eine Parameteranforderung, die eine Parameter-ID 74 enthält, an den Analyseausführungs-Eingangsdatenbehandler 110 senden, der wiederum die empfangene Parameter-ID 74 mit dem gespeicherten Parameteraufzeichnungszeiger in eine neue Parameteraufforderung korrelieren oder kombinieren wird, die dann an den Anrufsteuerdatenspeicherbehandler 118 gesendet wird.

Als ein sehr einfaches Beispiel, wie die Parameteranforderung an den Anrufsteuerdatenspeicherbehandler 118 zu bilden ist und ohne Einschränkung für die vielen möglichen Implementierungen, die einem Durchschnittsfachmann vertraut sind, kann der Parameteraufzeichnungszeiger die erste Stelle in dem Block 116 darstellen und die Parameter-ID 74 kann einen Versatz (offset) von dieser ersten Stelle darstellen, der zu dem Wert des Parameteraufzeichnungszeigers hinzugefügt wird. Der Parameterwert, der zu der Parameter-ID 74 in dem Anrufsteuerdatenblock 116 gehört, wird dann von dem Anrufsteuerdatenspeicherbehandler 118 durch den Analyseausführungs-Eingangsdatenbehandler 110 an die anfordernde Analyseprimitivfunktion 120 geleitet und in der Analyse verwendet. Eine andere Analyseprimitivfunktion 120 kann danach ein Zwischenergebnis oder Endergebnis erzeugen, welches an den Anrufsteuerdatenspeicherbehandler 118 gesendet und in dem Anrufsteuerdatenblock 116 gespeichert wird.

Es sei daran erinnert, dass sämtliche Anrufaufzeichnungen den gleichen Satz von Parametern enthalten, die von dem gleichen Satz von Parameter-IDs 74 identifiziert werden, aber das jede Anrufaufzeichnung Parameterwerte für einen bestimmten Anruf enthält, die den von den Parameterwerten in einer anderen Anrufaufzeichnung, die gerade gleichzeitig analysiert wird, ziemlich unterschiedlich sind. Der Parameter-Aufzeichnungszeiger unterscheidet deshalb den bestimmten Satz von Anrufparameterwerten, der analysiert werden soll. Die andere Information, die in den Eingangsdaten 44 enthalten ist, kann irgendwelche analyse-spezifische Information umfassen, die ebenfalls von den High-Level-Analyseprimitiven 120 (oder den High-Level-Analysemodulen 130) analysiert werden können.

Wie in Fig. 10 gezeigt werden der Parameteraufzeichnungszeiger und die andere Information, soweit vorhanden, in einem Analyseausführungs-Eingangsdatenbehandler 110 gespeichert, der mit einem Anrufsteuerdaten-Speicherbehandler 118 gekoppelt ist. Der Parameter-Aufzeichnungszeiger verweist auf oder stellt in irgendeiner Weise einen Bezug zu der Stelle eines Blocks von Anrufsteuerdaten für den Anruf, der gerade analysiert wird, 116 in der Anrufsteuerdatenbank 54, die in dem Anrufsteuerdaten-Speicherbehandler 118 gespeichert oder von diesem in irgendeiner anderen Weise behandelt wird, her. Der Anrufsteuerdatenblock 116 umfasst sämtliche relevante Daten für den Anruf, der gerade analysiert wird (jeder Anruf kann mehrere verschiedene Analysen beinhalten), d.h. Daten, die von einem Teilnehmer eingegeben werden, oder andere Anrufverarbeitungsergebnisse, die für diesen bestimmten Anruf gültig sind, sowie Daten, die sich auf die Teilnahmeberechtigung oder den Aufenthaltsort des Teilnehmers beziehen, der für sämtliche Anrufe von dem Teilnehmer gültig ist.

Auf den Anrufsteuerdatenspeicherbehandler 118 kann von dem Analyseausführungs-Eingangsdatenbehandler 110, der in Reaktion auf eine Aufforderung von der Do-Prozedur 112 einer High-

Es sei an dieser Stelle darauf hingewiesen, dass für die Zwecke der im Anhang A beschriebenen beispielhaften High-Level-Analyseprimitivfunktionen 120 angenommen wird, dass die Eingangsdaten 44 nur aus dem externen Namen der Analyse und dem Parameteraufzeichnungszeiger bestehen und es wird nicht angenommen, dass andere Information in den Eingangsdaten 44 enthalten ist. Demzufolge verwendet keine der beispielhaften Analyseprimitivfunktionen 120 im Anhang A irgendeine andere Information, die in den Eingangsdaten 44 enthalten sein könnte. Während dies im Anhang A oder Anhang C nicht explizit erwähnt ist, wird von Durchschnittsfachleuten erkannt, dass die Low-Level-Analyseprimitivfunktionen 120 und die High-Level-Analyseprimitivfunktionen 120 (oder äquivalent die High-Level-Analysemodule 130) der vorliegenden Erfindung leicht andere (kontext-spezifische) Information verwenden können, die in den Eingangsdaten 44 enthalten sein kann.

Zum Beispiel können die Low-Level-Analyseprimitive 120 die andere Information über den Parameterstapel verwenden, d.h. die kontext-spezifische Information, die von dem Anrufsteuerprogramm 56 bereitgestellt wird, kann oben auf dem Stapel als ein erster Stapelrahmen vor der Ausführung der in den Eingangsdaten 44 spezifizierten Analyse angeordnet werden. Die Analyse kann dann diese Information durch Steuerebenen-Analysemodule 130, die von High-Level-Analysemodulen (130) definiert werden, die rekursiv aus Low-Level-Analyseprimitiven 120 gebildet (gebaut) werden, verwenden. Eine Verwendung der anderen Information in der Routing-Analyse kann z.B. dafür vorgesehen sein, um die Ausführung einer Routing-Analyse (Wegleitungsanalyse) durch einen sekundären Wegleitungs-Pfad (Muster) in dem Netz anzuzeigen, wenn es bekannt ist, dass ein primärer Routing-Pfad zu der gleichen gewünschten Zielstelle verstopft ist (mittels einer vorherigen Ausführung einer Wegleitungs-Analyse durch den primären Pfad). Als ein anderes Beispiel kann die andere Information Zeiger zu den Anrufaufzeichnungen für andere Anrufe umfassen, die in ähnlicher Weise zu der Anrufaufzeichnung analysiert werden

sollen, auf die mit dem Parameteraufzeichnungszeiger verwiesen wird.

Als nächstes beziehend auf Fig. 11 umfasst die bevorzugte Ausführungsform der Analysesteuerdaten 42 ein Analysesteuerdaten-Wörterbuch 132, das den externen Namen einer Analyse, die von dem Anrufsteuerprogramm 56 befohlen wird, in einen Zeiger zu den Analysesteuerdaten für diese bestimmte Analyse umwandelt. Zwei spezifische Beispiele von externen Analysenamen, nämlich "Ziffernanalyse" und "Routing-Analyse" sind in Fig. 11 gezeigt (ein externer Name muss nicht auf eine Textkette beschränkt sein, sondern kann im allgemeinen irgendeine Bezeichnung oder ein Objekt sein, das extern zu dem allgemeinen Analysesystem erkannt wird, d.h. von der Betreiberschnittstelle 200 und der Anrufverarbeitungssoftware). Das Analysesteuerdaten-Wörterbuch 130 übersetzt "Ziffernanalyse" und "Routing-Analyse" und gibt Zeiger zu den entsprechenden Analysesteuerdaten zurück, nämlich Ziffernanalysesteuerdaten 134 bzw. Routinganalyse-Steuerdaten 136.

In der bevorzugten Ausführungsform ist der Aufbau der Analysesteuerdaten für irgendeine Analyse, z.B. Ziffernanalyse-Steuerdaten 134, ähnlich zu dem Aufbau von irgendeiner anderen Analyse, z.B. von Routinganalyse-Steuerdaten 136. Jede Analysesteuerdatenstruktur, z.B. die Routinganalyse-Steuerdaten 136, umfasst ein Analysemodul-Wörterbuch 138, das die externen Namen von Analysemodulen 130 in Modul-Token umwandelt, d.h. Zahlen, die indirekt die Analysemodule 130 innerhalb dieser bestimmten Analysesteuerdaten identifizieren. Gemäss der bevorzugten Ausführungsform weist jedes der Analysemodule 130 in einer Analysesteuerdatenstruktur einen externen Namen auf, der z.B. mit einer Voranstellung (Präfix) beginnen kann, die die bestimmte Analyse identifiziert, gefolgt von einer Nummer, z.B. RA275 (dieses Format ist nur beispielhaft und ein Betreiber kann irgendein anderes geeignetes Format, mit oder

ohne eine Voranstellung, wählen). Wie aus der Diskussion der Ausführungs- und Kompilierungs Zustände der vorliegenden Erfindung deutlich wird, wird das Analysemodulwörterbuch 138 beim Kompilieren (und nicht beim Ausführen) von Analysemodulen 130 verwendet.

Unter weiterer Bezugnahme auf die Figur 11 umfasst jede Analysesteuerdatenstruktur ferner einen Analyseausführungs-Startdatenwert 140, der ein Start-Token 142 und einen Tabellenzeiger 144 für aktive Token umfasst. Das Start-Token 142 stellt einen Zeiger zu dem Stamm des Analysebaums für eine bestimmte Analyse bereit, z.B. dem Startstauerebenen-Analysemodul 150 in der Routing-Analyse, während der Tabellenzeiger 144 für aktive Token einen Zeiger zu der "aktiven" Analysemodul-Token-tabelle 146 bereitstellt, der auf die Analysemodule 130 verweist, die zum Ausführen einer Analyse verwendet werden. Der Tabellenzeiger 144 für aktive Token ermöglicht die parallele Verwendung von mehreren Analysemodul-Token-tabellen einschließlich der aktiven Analysemodul-Token-tabelle 146, die zum Analysieren eines echten Verkehrs durch das Netz verwendet wird, und einer oder mehrerer "passiver" Analysemodul-Token-tabellen, z.B. der Analysemodul-Token-tabelle 148, die von dem Betreiber verwendet werden kann, um mehr Analysemodule 130 innerhalb der Analysesteuerdaten für eine bestimmte Analyse zu modifizieren, z.B. zu bilden, während die aktive Analysemodul-Token-tabelle 146 gerade zum Ausführen der tatsächlichen Analyse, d.h. zum Ausführen von Analysemodulen 130, verwendet wird.

Jede Analysemodul-Token-tabelle 146 und 148 wird von einem Modultoken indiziert, der auf die tatsächlichen Speicherstellen (Adressen) der Analysemodule 130 verweist. Wie mit dem Analyseprimitiv-Funktionstoken und den Analyse-Primitiven 120 ermöglicht die Verwendung des Modultokens, d.h. der indirekten Token-Threaded-Technik, anstelle einer Verweisung direkt mit einem Zeiger auf die Analysemodule 130, eine Neuordnung eines Speichers, z.B. eine Bewegung,

Modifizierung und Ersetzung von Analysemodulen 130 innerhalb eines Speichers. Insbesondere kann eine Modifikation der Analysesteuerdaten für eine bestimmte Analyse, die in der Reihenfolge von mehreren Megabites sein kann, indirekt durch Modifizieren der Zeiger (zu den Analysemodulen) in einer der passiven Analysemodul-Tokentabellen erzielt werden. Um eine bestimmte Analyse zu modifizieren, z.B. eine Routing-Analyse, kann der Betreiber einfach die Inhalte der aktiven Analysemodul-Tokentabelle 146 in die passive Analysemodul-Tokentabelle 148 kopieren (die Zeiger kopieren, anstelle die Analysemodule zu kopieren), wenigstens einige der Zeiger zu den Analysemodulen 130 modifizieren, die neue Analysestruktur testen und die Analysemodul-Tokentabelle 148 aktivieren, nachdem das Testen beendet worden ist (den Wert des aktiven Tokentabellenzeigers 144 ändern, so dass er auf die Analysemodul-Tokentabelle 148 verweist).

Gemäß Fig. 11 umfassen die Analysesteuerdaten für jede Analyse, z.B. Ziffernanalyse-Steuerdaten 134 oder Routinganalysesteuerdaten 136, einen Zeiger auf den Stamm der Analyse, d.h. das Starttoken 142 verweist auf den Startpunkt für die Analyse. Wenn erwünscht, kann jedoch jede Analyse mehr als einen möglichen Startpunkt haben, d.h. mehrere unterschiedliche mögliche Startsteuerebenen-Analysemodule 130 innerhalb der gleichen Analysesteuerdaten-Struktur. Für diesen Zweck kann das Starttoken 142 von den Analyseausführungs-Startdaten 140 zu dem Analysesteuerdaten-Wörterbuch 132 bewegt werden, um eine dritte Spalte so zu bilden, dass jeder Eintrag in dem Analysesteuerdaten-Wörterbuch 132 drei Attribute aufweisen wird, nämlich einen externen Namen, einen Analysesteuerdatenanzeiger und ein Starttoken. Diese Implementierung ermöglicht tatsächlich die Erzeugung von verschiedenen Analysen innerhalb der gleichen Analysesteuerdatenstruktur, wobei jede unterschiedliche Analyse von einem anderen externen Namen identifiziert wird, der sowohl in einem Zeiger zu den entsprechenden Analysesteuerdaten als auch in einen Zeiger zu dem

entsprechenden Startanalysemodul 130 in den entsprechenden Analysesteuerdaten umgesetzt wird. Zum Beispiel werden die externen Namen "Routing-Analyse-1", "Routing-Analyse-2" und "Routing-Analyse-3" auf die gleichen Analysesteuerdaten verweisen, nämlich auf die Routinganalyse-Steuerdaten 136, werden aber jeweils auf ein anderes Startanalysemodul 130 verweisen.

Unter Bezugnahme als nächstes auf Fig. 12 ist der Aufbau eines gemeinsamen Typs von Steuerebenen-Analysemodule 130, die zum Beispiel in der Routing-Analyse verwendet werden können, aus einem Analyse-Administrationsmodul-Anfangsblock (Header) 160, einem Primitivfunktions-Token 162, einem Parametergebiet 164 und einem nächsten Modulreferenzgebiet 166 (die Inhalte dieser Struktur werden als ein Satz von aufeinanderfolgenden Speicherstellen nur für den Zweck einer Verdeutlichung der Beschreibung konfiguriert) gebildet. Der Analyse-Administrationsmodul-Header 160 wird für administrative Zwecke verwendet und kann zum Beispiel den kompilierten Namen des Steuerebenen-Analysemoduls 130 speichern. Das Primitivfunktions-Token 162 verweist entweder auf ein High-Level-Analyseprimitiv 120 oder ein High-Level-Analysemodul 130. Während einer Ausführung wird das Primitivfunktions-Token 162 durch die Analyseprimitivfunktions-Tokentabelle 106 oder von der Analysemodul-Tokentabelle 146 der Routinganalyse-Steuerdaten 136 in einen Zeiger zu der Do-Prozedur des bestimmten Analyseprimitivs 120 umgesetzt, das diesen bestimmten Typ von Steuerebenen-Analysemodul 130 interpretiert.

Für das in Fig. 12 gezeigte Analysemodul 130 kann das Primitivfunktions-Token 162 auf ein High-Level-Analyseprimitiv 120 des Typs "Indexwähler" weisen und die Inhalte des Parametergebiets 164 und des nächsten Modulreferenzgebiets 166 können wie in Fig. 13 gezeigt sein. Das Parametergebiet 164 enthält eine Referenz zu dem Parameter oder dem Datenwert, der analysiert werden soll, z.B. speichert die Parameter-ID 74,

die auf einen Parameterwert in den Anrufsteuerdaten 54 zeigt, und einen maximalen Wert für den Parameter, auf den Bezug genommen wird (im allgemeinen kann das Parametergebiet 164 eine Kombination von Konstanten etc. enthalten und verweist auf Anrufdaten, in Abhängigkeit von der Funktion des Analyseprimitivs, auf das von dem Primitivfunktions-Token 162 Bezug genommen wird). Das nächste Modulreferenzgebiet 166 enthält eine Tabelle von Analysemodultokens, die von dem Indexwähler-Analyseprimitiv 120 indiziert wird, um ein Modultoken zu wählen, das wiederum von der Analysemodul-Token-tabelle 146 in einen Zeiger zu dem nächsten Analysemodul 154 umgesetzt wird.

Wie voranstehend diskutiert, hängen die spezifischen Inhalte und die Organisation von Information in dem Steuerfeld 60 von irgendeinem Steuerebenen-Analysemodul 130 von dem Typ des High-Level-Analyseprimitivs 120 (oder äquivalent dem Typ des High-Level-Analysemoduls 130), auf das von dem Primitivfunktions-Token 162 Bezug genommen wird, d.h. dem High-Level-Analyseprimitiv 120 (oder dem High-Level-Analysemodul 130), das zum Interpretieren des bestimmten Typs von Steuerebenen-Analysemodul 130 verwendet wird, ab. Die Fig. 12-13 zeigen den Fall, bei dem das Primitivfunktions-Token 162 auf ein Indexwähler-Analyseprimitiv 120 und ein Steuerfeld 60 verweist, deshalb aus dem Parametergebiet 164 und dem nächsten Modulreferenzgebiet 166 besteht. In anderen Fällen, bei denen das Primitivfunktions-Token 162 zum Beispiel auf einem Zirkularverteilungswähler des in Fig. 9A gezeigten Typs oder einen zeitabhängigen Wähler des in Fig. 9B gezeigten Typs verweist, speichert das Steuerfeld 60 nicht eine Parameter-ID 74 einer indizierten Tabelle von Analysemodultokens, sondern speichert anstelle davon andere Typen von Steuerdaten 76, z.B. in Fig. 9A den zuletzt verwendeten Auslass-Zirkularzähler 76b und Token zu anderen Analysemodulen 130, die alternativ als das nächste Analysemodul 130 gewählt werden können. Es sei darauf hingewiesen, dass viele Veränderungen des Aufbaus des Analysemoduls 130, das in Fig. 12 gezeigt ist, möglich sind

und das nur das Primitivfunktions-Token 162 ein notwendiges Element des Aufbaus jedes Steuerebenen-Analysemoduls 130 ist.

Ausführungszustand

In der bevorzugten Ausführungsform arbeitet das allgemeine Analysesystem der vorliegenden Erfindung in einem oder beiden von zwei Zuständen, nämlich einer Ausführung oder einer Kompilierung (Zusammenstellung). In dem Ausführungszustand verwendet die Analyseinterpretiererprozedur 108 die Do-Prozedur 112 der Analyseprimitivfunktionen 120, um eine Analyse auszuführen. Um dies zu verdeutlichen, sei angenommen, dass das Anrufsteuerprogramm 56 die Analyseinterpretiererprozedur 108 angewiesen hat, die Ausführung einer bestimmten Analyse zu beginnen. Wie voranstehend beschrieben wurde, sendet das Anrufsteuerprogramm 56 an die Analyseinterpretiererprozedur 108 die Eingangsdaten 44, die in der Analyse verwendet werden sollen. Die Eingangsdaten 44 umfassen den externen Namen der gewünschten Analyse, z.B. "Ziffernanalyse" oder "Routinganalyse", was den Start für die Analyse, d.h. die Bündelung des Analysebaums, spezifiziert. Die Eingangsdaten 44 umfassen auch einen Parameteraufzeichnungszeiger, der auf die Anrufsteuerdaten für einen Anruf, der gerade analysiert wird (Anrufaufzeichnung) 116, und möglicherweise andere Information, die für die bestimmte Analyse spezifisch ist.

Wie in Fig. 10 gezeigt, werden der Parameteraufzeichnungszeiger und die andere Information in dem Analyseausführungs-Eingangsdatenbehandler 110 gespeichert, der später verwendet werden soll. Der externe Name der Analyse wird andererseits an das Analysesteuerdaten-Wörterbuch 132 geleitet, das, wie in Fig. 11 gezeigt, eine Auflistung von sämtlichen Analysesteuerdaten enthält, die von dem Anrufsteuerprogramm 56 aufgerufen werden können. Eine Suche durch das Analysesteuerdaten-Wörterbuch 130 nach dem externen Namen, der von dem Anrufsteuerprogramm 56 gesendet wird, gibt

einen Zeiger zu den geeigneten Analysesteuerdaten für die gewünschte (benannte) Analyse zurück. Wenn zum Beispiel angenommen wird, dass das Anrufsteuerprogramm 56 eine Routinganalyse angefordert hat, z.B. den externen Namen "Routinganalyse" an die Analyseinterpretiererprozedur 108 gesendet hat, dann wird das Analysesteuerdaten-Wörterbuch 132 einen Zeiger zu den Routinganalyse-Steuerdaten 136 zurückgeben.

Wie in Fig. 11 gezeigt, umfassen die Routinganalyse-Steuerdaten 136 die Analyseausführungs-Startdaten 140, die sowohl das Starttoken 142 als auch den aktiven Tokentabellenzeiger 144 für eine Routinganalyse enthalten. Das Starttoken (Wert 28 in Fig. 11) wird verwendet, um die aktive Analysemodul-Tokentabelle 146 zu indizieren, die von dem aktiven Tokentabellenzeiger 144 angezeigt wird, um einen Zeiger zu dem ersten Steuerebenen-Analysemodul 140 bei der Routinganalyse, d.h. den Stamm des Routinganalysebaums, zu erhalten. Das Primitivfunktions-Token 162 in dem ersten Steuerebenen-Analysemodul 150 wird verwendet, um die Analyseprimitivfunktions-Tokentabelle 106 (oder die Analysemodul-Tokentabelle 146) zu indizieren, um einen Zeiger zu der Do-Prozedur 112 der High-Level-Analyseprimitivfunktion 120 (oder dem High-Level-Analysemodul 130) zu erhalten, die der bestimmten Struktur und Inhalten des Steuerfelds 60 des ersten Analysemoduls 150, z.B. dem Parametergebiet 164 und dem nächsten Modulreferenzgebiet 166 für ein Indexwähler-Typ Analysemodul 150, zugeordnet ist, d.h. diese interpretieren kann.

Es sei ferner angenommen, dass zum Beispiel das Primitivfunktions-Token 162 des ersten Steuerebenen-Analysemoduls 150 auf eine Indexwähler-Typ High-Level-Analyseprimitivfunktion 120 verweist, die Do-Prozedur 112 des Indexwähler-Analyseprimitivs 120 die Parameter-ID 74 aus dem Parametergebiet 164 des ersten Steuerebenen-Analysemoduls 150 liest und den Analyseausführungs-Eingangsdatenbehandler 110

aufruft, der wiederum eine Aufforderung an den Analysesteuerdaten-Speicherbehandler 118 zum Lesen des Parameterwerts, der zu der Parameter-ID 74 in den Analysesteuerdaten für den Anruf 116, gehört der gerade analysiert wird, sendet. Dieser Parameterwert wird an die Do-Prozedur 112 des Indexzähler-Analyseprimitivs 120 gesendet und davon verwendet, um den geeigneten Auslass von dem nächsten Modulreferenzgebiet 166 in dem ersten Analysemodul 150 zu wählen, d.h. das nächste Analysemodul 130, das nach der Ausführung des ersten Analysemoduls 150 ausgeführt werden soll. Das nächste Analysemodul 150 kann dann, wie in Fig. 13 dargestellt, ausgeführt werden. Zur Übersichtlichkeit wird auch angenommen, dass das zweite Analysemodul 152 auf eine Indexwähler-Typ Analyseprimitiv 120 verweist.

Als nächstes beziehend auf Fig. 13 läßt sich nun eine bildliche Darstellung der Ausführung des zweiten Analysemoduls 152, nachdem das erste Analysemodul 150 in Fig. 11 ausgeführt worden ist, ansehen. Wie oben beschrieben, wurde ein Zeiger (Token) zu dem zweiten Analysemodul 152 auf Grundlage des Ergebnisses der Analyse gewählt, die von dem ersten Analysemodul 150 in dem Routinganalyse-Baum durchgeführt wurde. Die Analyseinterpretiererprozedur 108 liest das Primitivfunktions-Token 162 in dem Analysemodul 152 und registriert zum Beispiel den Wert 93, der auf die Analyseprimitivfunktion 120 verweist, die mit "Indexwähler" benannt ist. Der Wert 93 wird in der Analyseprimitivfunktions-Token-tabelle 106 in einen Zeiger zu der Do-Prozedur 112 des Analyseprimitiv-Funktions-Indexwählers umgewandelt. Die Do-Prozedur 112 des Analyseprimitivfunktions-Indexwählers wird dann die Daten in das Parametergebiet 164 des Analysemoduls 152 lesen. Wie in Fig. 13 gezeigt, umfasst das Parametergebiet 164 des Analysemoduls 152 sowohl eine Parameter-ID 74 für den Parameter "Ursprung" als auch einen maximalen Wert, der für diese Analyse von "Ursprung" von dem Analysemodul 152 zugelassen ist. Wie sich kurz ansehen lässt, wählt der Analyseprimitivfunktions-Indexwähler einen Auslass von dem

nächsten Modulreferenzgebiet 166 des Analysemoduls 152 in Abhängigkeit von dem Wert des "Ursprungs-"Parameters in dem Anrufrdatenblock 116.

Im allgemeinen wird der Parameter "Ursprung" verwendet, um eine Gruppe von Teilnehmern zu identifizieren, die von der Vermittlung in der gleichen Weise für Anrufverarbeitungszwecke behandelt werden (es können verschiedene "Ursprung-"Parameter für eine Wegleitung, eine Abrechnung, etc.) vorhanden sein. Für Wegleitungs-Zwecke kann der Wert des Parameters "Ursprung" zum Beispiel nicht nur den Ortscode bezeichnen, der von dem nationalen Numerierungsplan dem Gebiet der zentralen Vermittlungsstelle zugewiesen ist, sondern auch denjenigen Teil des geographischen Gebiets, der von der zentralen Vermittlungsstelle bedient wird, in dem sich der A-Teilnehmer befindet, oder andere Typen von Ursprung, beispielsweise die Geschäftsgruppe, zu der der A-Teilnehmer gehört. Der maximal zulässige Wert für den Wegleitungs-"Ursprung"-Parameter hängt von dem Numerierungsplan, dem Typ der Teilnahmeberechtigung, etc. ab und wird in diesem Fall auf 7 gewählt (ein Wert des "Ursprung-"Parameters größer als 7 kann zum Beispiel entweder einen Fehler in der Verarbeitung oder alternativ, wenn die Werte 8 und 9 für Geschäfts-(PBX oder Centrex-Gruppen reserviert sind, einen Anruf von einem Geschäftsteilnehmer, der eine weitere Analyse von "Ursprung" in einem anderen Analysemodul erfordert, anzeigen).

Unter Fortsetzung der Bezugnahme auf Fig. 13 sendet die Do-Prozedur 112 des Analyseprimitivfunktions-Indexwählers den Parameter ID 74 für "Ursprung" an den Analyseausführungs-Eingangsdaten-Behandler 110, der wiederum den Anrufsteuerdaten-Speicherbehandler 118 auffordert, die "Ursprungs"daten in den Anrufsteuerdaten für den Anruf 116, der gerade analysiert wird, zu lesen. Der Parameterwert des "Ursprung"-Datenwerts, hier 4, wird dann an die Do-Prozedur 112 des Analyseprimitivfunktions-Indexwählers gesendet. Die Do-Prozedur 112 des Analyseprimitivfunktions-Indexwählers

greift dann auf den maximal zulässigen Wert von dem Parametergebiet 164 des Analysemoduls 152 zu und vergleicht den "Ursprung"-Parameterwert mit dem maximal zulässigen Wert.

Wenn der "Ursprung"-Parameterwert kleiner oder gleich zu dem maximal zulässigen Wert ist, dann verwendet die Do-Prozedur 112 des Analyseprimitivfunktions-Indexwählers den "Ursprung"-Parameterwert, um eine Tabelle in dem nächsten Modulreferenzgebiet 166 des Analysemoduls 152 zu indizieren und wählt den geeigneten Auslass von dem Analysemodul 152, d.h. das Analysemodultoken für das nächste Analysemodul zum Ausführen. Wenn jedoch der "Ursprung"-Parameterwert größer als 7 war (wegen eines Fehlers in der Verarbeitung oder der Anruf war zum Beispiel von einem Geschäftsteilnehmer), wird das Analysemodultoken für ausserhalb eines Bereichs gewählt und von der aktiven Analysemodul-Token-tabelle 146 in einen Zeiger auf ein vorgegebenes Steuerebenen-Analysemodul 130 übersetzt.

Da in diesem Fall der "Ursprung"-Parameterwert (4) kleiner als der maximal zulässige Wert (7) ist, wird das Analysemodultoken, das von dem Parameterwert (4) indiziert wird, gewählt und von der aktiven Analysemodul-Token-tabelle 146 in einen Zeiger auf das nächste Steuerebenen-Analysemodul 154 übersetzt. Das nächste Analysemodul 154 wird dann in einer ähnlichen Weise wie die Ausführung des Analysemoduls 152 ausgeführt. Die Analyse wird entlang dieser Zeilen fortgesetzt, wobei ein Analysemodul 130 sequentiell nach einem anderen Analysemodul 130 in dem Wegleitungs-Analysebaum ausgeführt wird. Einige der Analysemodule 130 in dem Wegleitungs-Analysebaum werden Zwischenergebnisse oder Endergebnisse der Analyse erzeugen (die Endergebnis-Analysemodule 130 werden Endpunkte in dem Analysebaum sein). Die Analysemodule 130, die ein Ergebnis erzeugen, werden von Ergebniserzeugnis-Analyseprimitiven 120 (oder Analysemodulen 130, die ein Ergebnis erzeugen) interpretiert und die Ergebnisse können an die Anrufaufzeichnung 116 in den Anrufsteuerdaten 54 zur Verwendung in einer oder mehreren

nachfolgenden Analysen oder beim Steuern von irgendeinem externen Prozess zurückgeführt werden.

Kompilierungszustand

In der bevorzugten Ausführungsform der vorliegenden Erfindung wird der Kompilierungszustand aktiviert, wenn der Betreiber die Einrichtungen der Betreiberschnittstelle 200 verwendet (die mit der Analyse-Bildungsprozedur 102 kommuniziert), um neue Analysemodule 130 zu erzeugen. Der Betreiber kann neue Analysemodule 130 erzeugen, die zu irgendeiner der drei verschiedenen voranstehend diskutierten Kategorien gehören: (1) Low-Level-Analysemodule, die Worte (Unterprogramme) umfassen, die von anderen Low-Level-Analysenmodulen und von High-Level-Analysenmodulen aufgerufen werden können, (ii) High-Level Analysemodule, die Wörter definieren (Build-Prozedur und Do-Prozedur), die wie die High-Level-Analyseprimitivfunktionen arbeiten und (iii) Steuerebenen-Analysenmodule, die mit anderen Steuerebenen-Analysenmodulen in einer Analysestruktur unter Verwendung der Analysemodultokens in ihren Steuerfeldern 60, z.B. den Analysemodultoken in dem nächsten Modulreferenzgebiet 166 des Analysemoduls 152, das in Fig. 13 gezeigt ist, verbunden werden können. Die Analysestruktur, die die neuen Steuerebenen-Analysenmodule 130 umfasst, kann (i) eine Analyse-Superstruktur sein, die eine vollständige Analyse bildet, z.B. eine Ziffernanalyse oder eine Wegleitungsanalyse, oder (ii) eine Analyse-Substruktur, die von mehreren Stellen in einer Analyse aufgerufen werden kann.

Eine Analyse-Substruktur kann zum Beispiel beim Wählen einer bestimmten Route für einen Anruf, der gerade verarbeitet wird, verwendet werden. Eine derartige Substruktur kann Aktionen umfassen, die jedesmal ausgeführt werden sollten, wenn die Route von verschiedenen Teilen in einer Wegleitungs-Analyse gewählt wird, zum Beispiel beim Sammeln von Statistiken über die Anruf-Wegleitung oder eine Regulierung des Verkehrsbetrags, der an eine bestimmte Route gerichtet ist.

Andere Substrukturen können verwendet werden, um Analysesteuerdaten darzustellen, die die gleichen in mehreren zentralen Vermittlungsstellen sind, und somit einmal von dem Betreiber erstellt und später über die Betreiberschnittstelle 200 an das allgemeine Analysesystem in jeder zentralen Vermittlungsstelle geliefert werden können. Derartige andere Unterstrukturen können verwendet werden, zum Beispiel bei der Wegleitungs-Analyse von Notrufnummern, z.B. der Nummer 911, oder von internationalen Anrufen.

In der bevorzugten Ausführungsform sind die Analyseprimitivfunktionen 120 eingebaut oder hart-codiert innerhalb des allgemeinen Analyseprogramms 40 und es handelt sich bei ihnen um Typen, die allgemein entweder anwendungsneutral, d.h. für eine Verwendung in verschiedenen Anwendungen vorgesehen, oder benutzerneutral, d.h. für eine Verwendung von verschiedenen Benutzern der gleichen grundlegenden Anwendung (z.B. Anrufverarbeitung) oder beides sind. Jedes der High-Level-Analyseprimitive 120 umfasst eine Build-Prozedur 114 und kann deshalb verwendet werden, um neue Steuerebenen-Analysemodule eines bestimmten Typs in einer Weise zusammenzustellen, die etwas analog zu der Verwendung eines eingebauten definierenden Worts in Forth zum Definieren von neuen Forth-Wörtern ist. Indem auch ein Satz von Low-Level-Analyseprimitivfunktionen 120 ähnlich zu einigen der eingebauten Wörter von "Forth, z.B. eine Prozedur, die ein Wort PROC definiert (ähnlich wie ":" in Forth) und die <BUILD und DOES> Prozeduren, eingebaut werden, kann die Betreiberschnittstelle 200 auch verwendet werden, um neue Low-Level- und High-Level-Analysemodule 130 zu erzeugen. Die High-Level-Analysemodule 130 können wiederum verwendet werden, um anwendungsspezifische oder benutzerspezifische Steuerebenen-Analysemodule 130 zusammenzustellen, die auf die speziellen Anforderungen der Anwendung oder die Wünsche des Benutzers zugeschnitten sind. In der Anrufverarbeitungsanwendung kann zum Beispiel der Vermittlungsstellenbetreiber neue Steuerebenen-Analysemodule 130 zusammenstellen und diese mit

Steuerdaten füllen, die die bestimmte Konfiguration des Netzes des Betreibers (Numerierungsplan etc.) beschreiben.

Die Low-Level- und High-Level-Analysemodule 130 sind nicht in das allgemeine Analyseprogramm eingebaut oder hart-codiert, sondern werden von dem Benutzer gebaut und in den Analysesteuerdaten 42 gespeichert. Die Low-Level-Analysemodule 130 werden rekursiv aus den Low-Level-Analyseprimitivfunktionen 120 in einer Weise definiert, die etwas analog zu der Definition von neuen Forth-Wörtern aus eingebauten Forth-Wörtern ist. Die Low-Level-Analysemodule 130 und die Low-Level-Analyseprimitivfunktionen 120 können wiederum verwendet werden, um neue High-Level-Analysemodule 130 in einer Weise zu definieren, die etwas analog zu der Definition der neuen Forth-definierenden Wörter aus benutzerdefinierten und eingebauten Forth-Wörtern ist. Die neuen High-Level-Analysemodule 130 werden genauso wie die High-Level-Analyseprimitivfunktionen 120 verwendet, um neue Steuerebenen-Analysemodule 130 in einer Weise zu definieren, die etwas analog zu der Definition der neuen Forth-Wörter aus benutzerdefinierten definierenden Wörtern ist. Jedes neue High-Level-Analysemodul 130 wird die <BUILD und DOES> Funktionen verwenden, um sowohl das Kompilierungszeit-Verhalten (entsprechend zu der Build-Prozedur 114) als auch das Ausführungszeit-Verhalten (entsprechend zu der Do-Prozedur 112) jedes neuen Steuerebenen-Analysemoduls 130, das davon definiert wird, zu spezifizieren.

Um eine neue anwendungsspezifische oder benutzerspezifische Analysesuperstruktur oder Substruktur zu bilden, können eine oder mehrere der eingebauten Low-Level-Analyseprimitivfunktionen 120, d.h. Wörter (Unterprogramme), zunächst verwendet werden, um neue Low-Level-Analysemodule 130, d.h. neu rekursiv definierte Unterprogramme (neue Versionen von <BUILDS oder DOES> zum Beispiel) zu erzeugen (zu definieren). Die eingebauten Low-Level-Analyseprimitivfunktionen 120 und die benutzerdefinierten Low-

Level-Analysemodule 130 können dann verwendet werden, um neue High-Level-Analysemodule 130, d.h. definierende Worte, zu definieren, die wiederum zusammen mit den eingebauten High-Level-Analyseprimitivfunktionen 120 verwendet werden können, um neue Steuerebenen-Analysemodule 130 in dem Analysebaum zu bilden (wie sich aus der nachstehenden Beschreibung ersehen lässt, endet die Ausführung der Low-Level- und High-Level-Analyseprimitive 120 oder der Analysemodule 130 normalerweise mit einem CALL RETURN (ANRUFRÜCKSPRUNG), während die Ausführung der Steuerebenen-Analysemodule 130 normalerweise mit einer GO TO Verbindung zu dem nächsten Steuerebenen-Analysemodul in dem Analysebaum endet).

Der folgende Text stellt Beispiele bereit, wie die Betreiberschnittstelle 200 in der bevorzugten Ausführungsform der vorliegenden Erfindung verwendet werden kann, um (i) neue Steuerebenen-Analysemodule 130 aus eingebauten High-Level-Analyseprimitivfunktionen 120 zusammenzustellen, (ii) neue Low-Level-Analysemodule 130 aus (eingebauten) Low-Level-Analyseprimitivfunktionen 120 zu definieren, (iii) neue High-Level-Analysemodule 130 aus Low-Level-Analyseprimitivfunktionen 120 und Low-Level-Analysemodulen 130 zu definieren (die Low-Level-Analyseprimitivfunktionen 120 und die Low-Level-Analysemodule 130 sind allgemein Unterprogramme, die zum Definieren entweder von Low-Level-Analysemodulen oder High-Level-Analysemodulen verwendet werden können), und (iv) neue Steuerebenen-Analysemodule 130 aus benutzerdefinierten High-Level-Analysemodulen 130 zusammenzustellen. In den nachfolgenden Beispielen werden die Wörter, die über die Betreiberschnittstelle 200 eingegeben werden, in Grossbuchstaben dargestellt, um diese von englischen Wörtern zu unterscheiden.

Allgemeiner Überblick der Kompilierung von Steuerebenen-Analysemodulen

Die Analyse-Bildungsprozedur 102 und die Bildungsprozedur 114 der eingebauten High-Level-Analyseprimitivfunktion 120 oder ein früher definiertes High-Level-Analysemodul 130 können in einer Tamdemverschaltung verwendet werden, um neue Steuerebenen-Analysemodule 130 innerhalb bestimmter Analysesteuerdaten, z.B. Wegleitungsanalyse-Steuerdaten 136, zu definieren. Im allgemeinen wird die Definition von jedem neuen Steuerebenen-Analysemodul 130 umfassen den Namen der Analysesteuerdaten, die der bestimmten Analyse entsprechen, z.B. der Wegleitungs-Analyse, in die das neue Steuerebenen-Analysemodul 130 eingefügt werden soll, und den Namen des definierenden Worts, das einer bestimmten High-Level-Analyseprimitivfunktion 120 oder einer High-Level-Analysemodul 130 entspricht, z.B. "Indexwähler", was ein allgemeines Skelett (eine Schablone) eines Typs (einer Klasse) von Steuerebenen-Analysemodule bereitstellt, von denen das neue Analysemodul 130 ein spezifisches Ereignis (Objekt) ist. Die Analyse-Bildungsprozedur 102 wird das Analysesteuerdaten-Wörterbuch 132 und das Analyseprimitivfunktions-Wörterbuch 104 oder das Analysemodul-Wörterbuch 130 verwenden, um die Analysesteuerdaten wie die Bildungsprozedur der High-Level-Analyseprimitivfunktion 120 oder eines High-Level-Analysemoduls 130, die jeweils in der Definition benannt sind, zu identifizieren.

Bei der Kompilierung von Steuerebenen-Analysemodulen 130 gibt der Betreiber Information ein, die in zwei Teile aufgeteilt werden kann (diese Unterteilung ist jedoch eine Frage einer Designwahl). Der erste Teil besteht aus Information (i), die von dem Betreiber für sämtliche Steuerebenen-Analysemodultypen (Familien) eingegeben wird, und zwar unabhängig von dem bestimmten Typ vom Steuerebenen-Analysemodul, das gerade definiert wird, und (ii) die von der Analyse-Bildungsprozedur 102 behandelt wird (eine Gruppe von Steuerebenen-

Analysemodulen sind von dem gleichen "Typ" oder der gleichen "Familie", wenn sie definiert werden oder aus der gleichen High-Level-Analyseprimitivfunktion 120 oder dem gleichen High-Level-Analysemodul 130 definiert oder gebaut (gebildet) werden. Diese Information umfasst den externen Namen der Analyse, den externen Namen des neuen Steuerebenen-Analysemoduls, bestimmte administrative Information und den externen Namen der High-Level-Analyseprimitivfunktion 120 oder des High-Level-Analysemoduls 130, die/das zum Zusammenstellen des neuen Steuerebenen-Analysemoduls verwendet wird. Die administrative Information und der externe Name der High-Level-Analyseprimitivfunktion 120 oder des High-Level-Analysemoduls 130 wird in den Analyseadministrationsmodul-Header 160 bzw. das Primitivfunktions-Token 162 des neuen Steuerebenen-Analysemoduls 130 kompiliert.

Der zweite Teil der Betreibereingabe besteht aus Information (i), die für einen bestimmten Steuerebenen-Analysemodultyp spezifisch ist, d.h. von dem Typ der High-Level-Analyseprimitivfunktion 120 oder dem High-Level-Analysemodul 130 abhängt, die/das zum Definieren des neuen Steuerebenen-Analysemoduls 130 verwendet wird, und (ii) die von der spezifischen Bildungsprozedur dieser Analyseprimitivfunktion 120 oder diesem Analysemodul 130 behandelt wird. Diese Information (bestimmte Parameterwerte) werden in das Steuerfeld 60 des neuen Steuerebenen-Analysemoduls (kompiliert) darin eingebaut. Für den exemplarischen Typ von Steuerebenen-Analysemodul 130, das in Fig. 12 gezeigt ist, umfasst diese Information die Parameter-ID 74, die in das Parametergebiet 164 kompiliert ist, und die externen Namen der nächsten Analysemodule (mit Ausnahme von denjenigen Steuerebenen-Analysemodulen 130, die Endpunkte in der Analyse sein werden und deshalb keinen haben werden), die in das nächste Modulreferenzgebiet 166 des neuen Steuerebenen-Analysemoduls 130 kompiliert werden.

Kompilierung von Steuerebenen-Analysemodulen aus High-Level-Analyseprimitivfunktionen

Zur Veranschaulichung sei der Fall betrachtet, bei dem ein Steuerebenen-Analysemodul des Typs "Indexwähler", das in Fig. 13 gezeigt ist, d.h. ein Steuerebenen-Analysemodul 130, auf das das Indexwähler-Analyseprimitiv 120 verweist, erzeugt wird. Der Betreiber kann das "Indexwähler"-Steuerebenen-Analysemodul 152 durch Ausgeben eines Befehls an die Analyse-Bildungsprozedur 102 (Fig. 10) über die Betreiberschnittstelle 200 erzeugen. Dieser Befehl wird die folgenden Teile enthalten:

1. Das Befehlswort zum Erzeugen eines neuen Analysemoduls.
2. Den externen Namen der Analysesteuerdaten, in denen das neue Analysemodul liegen wird, z.B. "Routing-Analyse" (für die Routinganalyse-Steuerdaten 136 in Fig. 11).
3. Den externen Namen des Analysemoduls, das erzeugt werden soll, z.B. "RA 275".
4. Administrative Information, z.B. "Ursprung für eine Wegleitung", die in den Analysemodul-Administrationsheader 160 des neu erzeugten Analysemoduls 152 zusammengestellt wird.
5. Den externen Namen der Analyseprimitivfunktion 120, z.B. "Indexwähler", die beim Kompilieren (und Ausführen) des neuen Analysemoduls verwendet werden soll.
6. Information, die für die gewählte Analyseprimitivfunktion spezifisch ist. In diesem Beispiel (Indexwähler) ist die folgende Information erforderlich:

- a. Die Parameter-ID 74 für den Analysesteuerdatenparameter, der analysiert werden soll, z.B. die ID 74 für "Ursprung" (hier Ursprung für Wegleitungs-Zwecke). Diese Parameter-ID 74 kann ein Tokenwert sein oder kann alternativ ein symbolischer externer Name sein, der in einen Tokenwert-Verwendung einer Katalogfunktion umgewandelt wird.
- b. Eine Anzeige über die Gesamtanzahl von möglichen Auslassen von dem neuen Analysemodul, d.h. den höchsten (maximalen) Indexwert des analysierten Parameters, für den ein neues Analysemodul zugewiesen werden sollte.
- c. Für jeden Auslass (einschließlich des Auslasses ausserhalb-des Bereichs) den externen Namen des Analysemoduls, das als das nächste Analysemodul für diesen Auslass zugewiesen werden sollte.

In der bevorzugten Ausführungsform der Erfindung handelt es sich bei der Betreiberschnittstelle 200 um eine Computer-Computer-Schnittstelle, die gemäß dem OSI Modell konfiguriert ist und der eine beliebige Syntax gegeben werden kann. Die folgende Syntax ist ein Beispiel, wie die Syntax für einen Betreiberbefehl zum Erzeugen eines "Indexwähler-"Analysemoduls gebildet werden kann:

```
CREATE ROUTINGANALYSIS :RA275 "ORIGIN FOR
ROUTING" INDEX_SELECTOR(58, 4, RA187,
RA198, RA0, RA99, RA99, RA33)
```

Dieser Befehl (Textkette) von dem Betreiber wird von der Analyse-Bildungsprozedur 102 und der Bildungsprozedur 114 der Analyseprimitivfunktion "Indexwähler" in den folgenden Schritten gelesen und ausgewertet (interpretiert).

1. Die Analyse-Bildungsprozedur 102 erkennt den Befehl zum Erzeugen eines neuen Analysemoduls, d.h. das Schlüsselwort "Create" (Teil 1) und aktiviert die geeigneten inneren Teile, die die "Create"-Funktion ausführen. Andere Befehle, die die Analyse-Bildungsprozedur 102 erkennen kann, umfassen Befehle zum Modifizieren von Parameterwerten in (Inhalten von) existierenden Analysemodulen 130 und Befehle zum Ändern der aktiven Analysemodul-Token-tabelle 146 in den Analysesteuerdaten 136, z.B. zum Modifizieren des Werts des aktiven Token-tabellenzeigers 144, um anstelle davon auf die Analysemodul-Token-tabelle 148 zu verweisen.
2. Die aktivierten Teile der Analyse-Bildungsprozedur 102 extrahieren den externen Namen der Analyse, d.h. "Routing-Analyse" (Teil 2) und verwenden dann das Analysesteuerdaten-Wörterbuch 132 (das eine Auflistung von früher definierter Analyse enthält), um "Routing-Analyse" in einen Zeiger zu den Analysesteuerdaten 136 zu übersetzen.
3. Die aktivierten Teile der Analyse-Bildungsprozedur 102 extrahieren nun den externen Namen des neuen Analysemoduls, d.h. "RA 275" (Teil 3). Der externe Name des neuen Analysemoduls wird in das Analysemodul-Wörterbuch 138 eingegeben und ein nicht verwendetes Modultoken, z.B. der Wert 65, wird dem neuen Analysemodul zugewiesen.
4. Die aktivierten Teile der Analyse-Bildungsprozedur 102 extrahieren den externen Namen der Analyseprimitivfunktion, d.h. "Indexwähler" (Teil 5) und verwenden dann das Analyseprimitivfunktions-Wörterbuch 104 (das eine Auflistung von früher definierten Analyseprimitivfunktionen 120 enthält), um "Indexwähler" in ein Primitivfunktions-Token umzusetzen, das auf diese Analyseprimitivfunktion verweist, z.B. den Wert 93.

5. Die aktivierten Teile der Analyse-Bildungsprozedur 102 verwenden die Analyseprimitivfunktions-Token-tabelle 106, um das (im Schritt 4 gefundene) Primitivfunktions-Token in einen Zeiger zu der Build-Prozedur 114 des Analyseprimitivfunktions-Indexwählers zu übersetzen und dann eine Ausführung dieser Build-Prozedur 114 zu beginnen. An diesem Punkt geht die Steuerung von der Analysebildungsprozedur 102 an die Bildungs-Prozedur 114 des Indexwählers.
6. Die Bildungs-Prozedur 114 des Analyseprimitivfunktions-Indexwählers extrahiert die Information, die für den Indexwähler (Teil 6) spezifisch ist, weist einen Speicherplatz für das neue Indexwähler-Analysemodul in der Form der Fig. 12 zu und füllt das Parametergebiet 164 und das Modulreferenzgebiet 166 mit den geeigneten Werten in den folgenden Schritten:
 - a. Die Bildungsprozedur 114 des Analyseprimitivfunktions-Indexwählers extrahiert die Parameter-ID 74 des Analysesteuerdatenparameters, der analysiert werden soll, d.h. den Tokenwert (58) für "Ursprung" (Teil 6a) und den maximalen Wert des Anrufparameters, d.h. den Wert 4 (Teil 6b), aus denen die Anzahl von Auslässen von dem neuen Analysemodul berechnet werden können (die Anzahl von in dem nächsten Modulreferenzgebiet 166 zu reservierenden Sätzen ist um 2 größer als der maximale Wert des Anrufparameters, d.h. sechs Plätze hier, einer jeweils für die Werte 0-4 und ein anderer für außerhalb des Bereichs). An diesem Punkt wird die benötigte Größe des neuen Analysemoduls berechnet, ein leeres Analysemodul mit der geeigneten Größe wird zugewiesen und die extrahierte Information wird in das Parametergebiet 164 geschrieben.

- b. Für jeden Auslass (einschließlich des Auslasses für außerhalb-des-Bereichs), das in dem Eingabetextstrom genannt wird, extrahiert die Bildungsprozedur 114 des Analyseprimitivfunktions-Indexwählers den externen Namen des Analysemoduls (36c) und verwendet das Analysemodul-Wörterbuch 138 (welches eine Auflistung von früher definierten Analysemodulen 130 enthält), um das entsprechende Modultoken aufzufinden. Dieses Modultoken wird in dem geeigneten Schlitz in dem nächsten Modulreferenzgebiet 166 angeordnet. An diesem Punkt endet die Ausführung der Bildungsprozedur 114 des Analyseprimitivfunktions-Indexwählers und die Steuerung wird zurückgegeben und ein Zeiger auf das neue Analysemodul wird an die anfänglich aktivierten Teile der Analysebildungsprozedur 102 zurückgegeben.
7. Die aktivierten Teile der Analysebildungsprozedur 102 schreiben den zurückgegebenen Zeiger in das neue Analysemodul und das Modultoken (welches im Schritt 3 gefunden wird) in die inaktive (passive) Analysemodul-Token-tabelle 148, d.h. der Zeiger wird in einen Schlitz 65 in der passiven Analysemodul-Token-tabelle 148 geschrieben. Das neue Analysemodul wird dadurch an die Routinganalyse-Steuerdaten 136 "gebunden".
8. Die aktivierten Teile der Analysebildungsprozedur 102 schreiben das Primitivfunktionstoken, d.h. den Wert 93 (der im Schritt 4 gefunden wird) in das Primitivfunktions-Tokenfeld 162 des neuen Analysemoduls und schreiben auch die Administrationsinformation, d.h. "Ursprung für eine Wegleitung" (Teil 4) in den Administrations-Header 160 des neuen Analysemoduls. Der Analyseadministrations-Modulheader 160 wird deshalb eine kurze Beschreibung des neuen Analysemoduls ("Ursprung für eine Wegleitung") enthalten und das Primitivfunktions-

Token 162 wird einen Zeiger zu der Do-Prozedur 112 des Analyseprimitivfunktions-Indexwählers indizieren, der ausgeführt wird, wenn das neue Analysemodul ("RA 275") analysiert (interpretiert wird).

Einmal definiert kann das neue Analysemodul RA 25 bei der Wegleitungs-Analyse durch Kopieren des entsprechenden Modultokens und des Zeigers von der inaktiven Analysemodul-Token-tabelle 148 (Schritt 7) zu der aktiven Analysemodul-Token-tabelle 146 verwendet werden. Wenn RA 275 ausgeführt wird, dann wird die Do-Prozedur 112 des Indexwählers der Analyseprimitivfunktion 120 angezeigt von dem Primitivfunktionstoken 162 ausgeführt. Die Do-Prozedur 112 des Indexwählers wird einfach die Tabelle von Analysemodul-Tokens in dem nächsten Modulreferenzgebiet 166 indizieren und das Analysemodul-Token entsprechend zu dem Wert des "Ursprung"Anrufparameters wählen. Wenn z.B. der Wert von "Ursprung" 0 ist, dann wird RA 187 gewählt; wenn der Wert von "Ursprung" 1 ist, dann wird RA 198 gewählt; use. Somit wird die Wegleitungs-Analyse einem der Analysemodul-Token in dem nächsten Modulreferenzgebiet 166 von RA 275 zu dem nächsten Analysemodul 130 in dem Anaysebaum folgen.

Es sei darauf hingewiesen, dass der Zeiger zu dem neuen Analysemodul 130 zunächst in die passive Anaylsemodul-Token-tabelle 148 eingefügt wurde und dann die aktive Analysemodul-Token-tabelle 146 transferiert wurde. Allgemein werden sämtliche Modifikationen zu den Wegeleitungsanalyse-Steuerdaten 136 der passiven Analysemodul-Token-tabelle 148 durchgeführt. In der Praxis wird der Betreiber wünschen, eine Analysesuperstruktur oder Substruktur zu erzeugen oder zu modifizieren, die mehrere Steuerebenen-Analysemodule 130 umfassen, anstelle von nur einem Steuerebenen-Analysemodul 130 zu jeder Zeit. Für diesen Zweck werden die Inhalte der aktiven Analysemodul-Token-tabelle 146 zunächst in die passive Analysemodul-Token-tabelle 148 kopiert. Dieser Schritt erlaubt, dass die aktive Analysemodul-Token-tabelle 146 fortwährend in

der Routing-Analyse des Telefonverkehrs verwendet wird, während die Analysesuperstruktur oder Substruktur gerade in der passiven Analysemodul-Token-tabelle 148 modifiziert wird, und stellt ferner sicher, dass diese Token-tabellen intern konsistent miteinander sind.

Die Modifikationen an den Analysesteuerdaten können z.B. aus einer Reihe von Definitionen von neuen Analysemodulen 130, zu denen Zeiger in die Analysemodul-Token-tabelle 148 eingefügt werden, oder aus einer Reihe von Modifikationen an den Parameterwerten in existierenden Analysemodulen 130, die effektiv eine neue Analysesuperstruktur oder Substruktur erzeugen, bestehen. Sobald sie modifiziert ist, kann die Analysesuperstruktur oder Substruktur nach Fehlern geprüft werden, z.B. nach Endlosschleifen, offenen Enden etc. und wird dann dem Verkehr übergeben, indem der Wert des aktiven Token-tabellenzeigers geändert wird, um anstelle auf die Analysemodul-Token-tabelle 146 auf die Analysemodul-Token-tabelle 148 zu verweisen. Die vorher aktive Analysemodul-Token-tabelle 146 kann für eine vorgegebene Zeitperiode, z.B. 24 Stunden, gespeichert werden, bis die Effizienz der neuen aktiven Analysemodul-Token-tabelle 148 beurteilt werden kann. Wenn sich Verkehrsstörungen entwickeln sollten, kann die Analysemodul-Token-tabelle 146, die auf einen erwiesenen Satz von Analysemodulen 130 verweist, dem Verkehr neu übergeben (neu aktiviert) werden.

Im Anhang A wird ein Satz von High-Level-Analyseprimitivfunktionen 120 beschreiben, die beim Zusammenstellen von neuen Steuerebenen-Analysemodulen verwendet werden können. Dieser Satz ist dafür vorgesehen, um beispielhaft die Typen von High-Level-Funktionen, die innerhalb der bevorzugten Ausführungsform der vorliegenden Erfindung verwendet werden können, weiter darzustellen, d.h. die Anwendung des allgemeinen Analysesystems für die Steuerung einer Anrufverarbeitung in einem Telekommunikationssystem, z.B. einer zentralen Vermittlungsstelle.

Durchschnittsfachleute werden erkennen, dass viele andere High-Level-Analyseprimitivfunktionen implementiert werden können:

Kompilierung von Low-Level-Analysemodulen

Die folgende Zeile einer Eingabe zu den allgemeinen Analysesystemen illustriert die Verwendung von einigen der Low-Level-Analyseprimitivfunktionen 120, die im Anhang C beschrieben sind, um ein neues Low-Level-Analysemodul zu definieren:

```
PROC 2* DUPLICATE ADD ENDPROC
```

Der erste Teil (PROC) und der letzte Teil (ENDPROC) sind Low-Level-Analyseprimitivfunktionen 120, die zum Bilden von neuen Low-Level-Analysemodulen 130 in einer Weise verwendet werden, die etwas analog zu der Verwendung ":" und ";" in Forth ist. Jedes neue Low-Level-Analysemodul 130, das von PROC definiert wird, wird die kompilierte Definition enthalten, d.h. die kompilierten Wörter zwischen PROC und ENDPROC (2*, DUPLICAT und ADD). Der zweite Teil (2*) ist der externe Name für das neue Low-Level-Analysemodul 130. Die dritten und vierten Teile (DUPLICATE und ADD) sind Low-Level-Analyseprimitivfunktionen 120, die die Funktion des neuen Low-Level-Analysemoduls 130 implementieren. In diesem Beispiel soll die Funktion den Wert des Teils oben auf dem Ausführungszeitstapel mit zweimal dem Wert ersetzen (der Ausdruck "Stapel" bezieht sich auf eine sequentielle Liste von Teilen, bei der Teile angefügt oder entfernt werden können, so dass der entfernte Wert der an den Stapel zuletzt angehängte ist, d.h. wie in diesem technischen Gebiet altbekannt, wird dies als zuletzt-herein-zuerst-heraus oder LIFO-Stapel bezeichnet). Der Effekt des Stapels von der Ausführung der Low-Level-Analyseprimitivfunktionen DUPLICATE und ADD wird mit näheren Einzelheiten im Anhang C beschrieben.

Das neu-erzeugte Low-Level-Analysemoduls 130a, welches 2* genannt wird, ist in Fig. 14 gezeigt. Das Primitfunktions-Token 162 verweist auf einen Eintrag für die Low-Level-Analyseprimitivfunktionen PROC 120a in der Analyseprimitivfunktions-Token-tabelle 106. Das Steuerfeld 60, das von dem Bildungsteil der PROC-Analyseprimitivfunktion 120a kompiliert wird, besteht aus einer Sequenz von Funktionstokens, die auf die Einträge für die Low-Level-Analyseprimitivfunktionen DUPLICATE 120c, ADD 120d und ENDPROC 120b in der Analyseprimitivfunktions-Token-tabelle 106 verweisen. Dem neuen Low-Level-Analysemoduls 130a wird ein Analysemodul-Tokenwert zugewiesen und es wird in die Analysesteuerdaten so eingefügt werden, dass auf es mit dem externen Namen (2*) Bezug genommen werden kann, wenn andere Analysemodule kompiliert werden, und mit dem zugewiesenen Analysemodul-Token, wenn eine Analyse ausgeführt wird. Mit anderen Worten kann jedes danach kompilierte Wort das Wort 2* aufrufen, als ob es ein anderes Wort wäre (ein Wort wird durch einfaches Verwenden seines Namens in der Eingangsdefinition aufgerufen). Wenn aufgerufen, führt 2* seine Funktion aus und kehrt dann zurück (dies ist analog zu der Ausführung eines Unterprogramm-Aufrufs).

Kompilierung von High-Level-Analysemodulen

Die folgenden Teile einer Eingabe an dem allgemeinen Analysesystem illustrieren die Verwendung von einigen der Low-Level-Analyseprimitivfunktionen 120, die im Anhang C beschrieben sind, und des Low-Level-Analysemoduls 2*, das oben definiert ist, um ein neues High-Level-Analysemodul 130 zu definieren, das zum Kompilieren und Ausführen von Steuerebenen-Analysemodulen innerhalb eines Analysebaums verwendet werden kann.

EINGABE

PROC 4SECTOR
<BUILDS

FUNKTION UND BETRIEB

-- Startet Kompilierungs-

```

                                zeitvorgänge
0                                --- Startet einen Versatz
                                innerhalb des neuen
                                Analysemoduls.
INSERT_CALLPARM                -- Fügt eine Parameter ID 74 für
                                den Anrufparameter ein.
INSERT_MODULE                  -- Fügt Analysemodul-Token für einen
                                Wert "0" ein.
INSERT_MODULE                  -- dito für den Wert = 1.
INSERT_MODULE                  -- dito für den Wert = 2.
INSERT_MODULE                  -- dito für den Wert = 3.
DOES>                          -- Startet Ausführungszeitvorgänge.
0                                -- Der Versatz zu der Parameter-ID 74
                                in dem Analysemodul 130.
FETCH_PARAMETER                -- Greift auf den Anrufsteuerdaten-
                                Speicherbehandler 118 durch den
                                Analyseausführungs-Eingangsdaten-
                                Behandler 110 zu und drückt den
                                Wert des Parameters für den Anruf
                                116, der gerade analysiert wird,
                                auf den Stapel.
1 ADD 2*                        -- Berechnet den Versatz innerhalb
                                des Analysemoduls zu dem Auslass
                                für den Wert (es wird angenommen,
                                dass das Analysemodul-Token 2
                                Bytes lang ist).
PERFORM                        -- Liest das Analysemodul-Token an
                                dem Versatz, der von dem Wert
                                oben an dem Stapel angezeigt
                                wird, und führt einen Sprung zu
                                diesem Analysemodul aus.
ENDPROC

```

Die Funktion der Analysemodule, die mit dem definierenden Analysemodul 4SELECTOR zusammengestellt sind, soll einen Auslass in Abhängigkeit von dem Wert in dem Bereich 0-3 eines Anrufparameters in den Anrufsteuerdaten 116 wählen (das High-

Level-Analysemodul 4SELECTOR mit 4 Auslässen arbeitet grösstenteils genauso wie die High-Level-Analyseprimitivfunktion SELECTOR, mit zehn Auslässen, die voranstehend im Zusammenhang mit Fig. 8 diskutiert wurde). Wie mit der Definition von 2* wird die Definition von 4SELECTOR innerhalb der Low-Level-Analyseprimitivfunktionen PROC und ENDPROC eingeschlossen. Die Definition des 4SELECTOR wird jedoch in eine Bildungsprozedur, die innerhalb der Low-Level-Analyseprimitivfunktionen <BUILD und DOES> eingeschlossen ist, und eine DO-Prozedur, die innerhalb der Low-Level-Analyseprimitivfunktionen DOES> und ENDPROC eingeschlossen ist, aufgeteilt.

Die Bildungsprozedur von 4SELECTOR besteht aus einer Sequenz von Low-Level-Analyseprimitivfunktionen 120 von früher definierten Low-Level-Analysemodulen 130, die spezifizieren, wie die Analysemodule 130 des 4SELECTOR-Typs zusammengestellt (kompiliert) werden, d.h. sie spezifizieren die Operationen, die die Inhalte des Steuerfelds 60 des erzeugten Analysemoduls 130 schreiben (für den Fall von 4SELECTOR wird das Steuerfeld 60 aus dem Parametergebiet 164 und dem nächsten Modulreferenzgebiet 166 bestehen, wie in Fig. 12 gezeigt). Die Low-Level-Analyseprimitivfunktionen, die innerhalb der Bildungssequenz verwendet werden, sind mit näheren Einzelheiten im Anhang C beschrieben.

Die Do-Prozedur von 4SELECTOR besteht aus einer Sequenz von Low-Level-Analyseprimitivfunktionen 120 oder früher definierten Low-Level-Analysemodulen 130, die spezifizieren, wie die Analysemodule 130 des 4SELECTOR-Typs ausgeführt werden, d.h. sie spezifizieren die Operationen, die die Inhalte des Parametergebiets 164 und des nächsten Modulreferenzgebiets 166 verwenden, um das nächste Analysemodul 130 in dem Analysebaum zu wählen. Die Low-Level-Analyseprimitivfunktionen 120 (FETCH_PARAMETER, 0, 1, ADD und PERFORM), die innerhalb der Do-Sequenz verwendet werden, sind im Anhang C beschrieben, während 2* das voranstehend beschriebene Low-Level-Analysemodul ist.

Das neu-erzeugte High-Level-Analysemodule 130b, welches 4SELECTOR genannt ist, ist in Fig. 15 dargestellt. Das Primitivfunktions-Token 162 bezieht sich auf einen Eintrag für die Low-Level-Analyseprimitivfunktion PROC 120a in der Analyseprimitivfunktions-Token-tabelle 106. Das Steuerfeld 60, welches durch den Bildungsteil der PROC 120a zusammengestellt wurde, besteht zunächst aus einem Abschnitt, der rekursiv von dem Bildungsteil der <BUILD-Prozedur zusammengestellt wird und sowohl einen Zeiger auf die Do-Prozedur als auch ein Funktionstoken enthält, das auf die Analyseprimitivfunktion <BUILD 120e verweist. Die übrigen Abschnitte des Parametergebiets 160, das von dem Bildungsteil der PROC Analyseprimitivfunktion zusammengestellt wird, enthält eine Sequenz von Funktionstoken, die sich auf die Analyseprimitivfunktionen 0 (die Konstante "0") 120f, INSERT_CALLPARAM 120g, INSERT_MODULE 120h, DOES>120i, FETCH_PARAMETER 120j, 1 (die Konstante "1") 120k, ADD 120l (120d in Fig. 14), PERFORM 120m und ENDPROC 120b und das Low-Level-Analysemodul 130a für 2* beziehen. Dem neuen High-Level-Analysemodul 130b wird ein Analysemodul-Tokenwert zugewiesen und es wird in die Analysesteuerdaten so eingefügt, dass auf es mit dem externen Namen "4SELECTOR" Bezug genommen werden kann, wenn andere Analysemodule kompiliert werden, und mit dem zugewiesenen Analysemodultoken, wenn eine Analyse ausgeführt wird, z.B. eine Routing-Analyse.

Einmal definiert kann das High-Level-Analysemodul 4SELECTOR verwendet werden, um neue Steuerebenen-Analysemodule 130 zusammenzustellen. Bei der Kompilierungszeit, d.h. wenn die Bildungsprozedur des 4SELECTOR ausgeführt wird, wird INSERT_CALLPARAM in das neue Analysemodul 130, das gerade kompiliert wird, einen Zeiger zu einem Anrufparameter einfügen, d.h. kompiliert den externen Namen des Anrufparameters, der analysiert werden soll, in eine Parameter-ID 74, die auf die Stelle dieses Anrufparameters in der Anrufaufzeichnung 116 verweist und INSERT_MODUL (auf welches in der Sequenz viermal

Bezug genommen wird) wird in das Analysemodul 130, welches kompiliert wird, eine Sequenz von vier Modulanzeigern 80 (entsprechend der vier Anrufparameterwerte in dem Bereich 0 bis 3) einfügen, die das Analysemodul 130, welches gerade kompiliert wird, mit den nächsten Analysemodulen in dem Analysebaum verbindet. Bei der Ausführungszeit, d.h. wenn die Do-Prozedur von 4SELECTOR ausgeführt wird, wird FETCH_PARAMETER den Parameter 46, z.B. eine gewählte Ziffer, angezeigt von der Parameter-ID 74 aus den Anrufsteuerdaten 54, holen und die Sequenz (1 ADD 2* PERFORM) wird für eine Ausführung das nächste Analysemodul 130 wählen, welches von dem Modulanzeiger 80 angezeigt wird, der zu dem Wert (0-3) des Parameters 46 in dem 4SELECTOR-Typ Analysemodul 130, das gerade interpretiert wird, gehört.

Kompilierung von Steuerebenen-Analysemodulen aus High-Level-Analysemodulen

Die folgenden Teile einer Eingabe in das allgemeine Analysesystem illustrieren die Verwendung des High-Level-Analysemoduls 4SELECTOR um ein neues Steuerebenen-Analysemodul 130 in einem Analysebaum zu definieren (zu kompilieren), das z.B. in den Routing-Analyse-Steuerdaten 136 liegt:

EINGABE

4SELECTOR THISID

FUNKTION UND BETRIEB

-- Befiehlt dem allgemeinen Analysesystem, ein neues Steuerebenen-Analysemodul mit einem externen Namen THISID des Typs 4SELECTOR zu kompilieren.

LINECLASS

-- Kompilierungszeit-Eingabe zu dem INSERT_CALLPARM in der Bildungsprozedur von 4SELECTOR. INSERT_CALLPARM wird die Parameter-ID 74 für den

Anrufparameter LINECLASS in das neue Analysemodul einfügen.

MODULEID_0

-- Kompilierungszeiteingabe zu dem ersten INSERT_MODULE in der Bildungsprozedur von 4SELECTOR. INSERT_MODULE wird das Analysemodultoken für das nächste Analysemodul in dem Analysemodulbaum, das gewählt werden soll, einfügen, wenn der Wert von LINECLASS 0 (Null) ist.

MODULEID_1

-- Dito für das zweite INSERT_MODULE und für den Wert = 1.

MODULEID_2

-- Dito für das dritte INSERT_MODULE und für den Wert = 2.

MODULEID_3

-- Dito für das vierte INSERT_MODULE und den Wert = 3.

Der Aufbau des Analysemoduls, welches durch diese Eingabezeilen erzeugt wird, ist in Fig. 16 dargestellt. Das neue Analysemodul 130c des 4SELECTOR Typs enthält den Wert eines Primitivfunktionstokens 162, das sich auf das Definieren der Analysemodul 130b mit Namen 4SELECTOR über die Analysemodul-Token-tabelle 146 bezieht. Das Parametergebiet 164 des Analysemoduls 130c enthält eine Parameter-ID 74, die sich auf den Anrufparameter LINECLASS in den Anrufsteuerdaten 116 bezieht. Das nächste Modulreferenzgebiet 166 des Analysemoduls 130c enthält eine Tabelle von vier Analysemodultokens, die über die Analysemodul-Token-tabelle 146, sich auf die nächsten Analysemodule 130d, 130e, 130f und 130g beziehen, die für Anrufparameter-werte 0, 1, 2 bzw. 3 gewählt werden sollen.

Erweiterbarkeit

Somit können genauso wie neue Steuerebenen-Analysemodule 130 aus eingebauten High-Level-Analyseprimitivfunktionen 120 definiert und zu den Analysesteuerdaten für eine bestimmte

Analyse hinzugefügt werden können, z.B. können die Routing-Analyse-Steuerdaten 136, neue Typen von Steuerebenen, neue Typen von Steuerebenen-Analysemodule aus benutzer-definierten High-Level-Analysemodulen 130 gebildet werden. Z.B. könnte in der vorangehenden Beschreibung, wie ein Indexwähler-Typ Analysemodul erzeugt wird, die Bildungsprozedur des High-Level-Analysemoduls 130 anstelle einer High-Level-Analyseprimitivfunktion 120 verwendet werden. Das High-Level-Analysemodul 130, das wie eine High-Level-Analyseprimitivfunktion 120 arbeitet, kann aus eingebauten Low-Level-Analyseprimitiven 120 und aus früher definierten Low-Level-Analysemodulen 130 gebildet werden, die wiederum rekursiv aus den Low-Level-Analyseprimitiven 120 definiert wurden. Diese Erweiterbarkeit des allgemeinen Analysesystems ermöglicht dem Benutzer die Freiheit, anwendungs-spezifische oder benutzer-spezifische Analysen in die Analysesteuerdaten 42 zu bauen, anstelle einer Lieferung einer neuen Version des allgemeinen Analyseprogramms zu nehmen, das diese bestimmten Anforderungen durch spezielle Typen von eingebauten Funktionen erfüllt.

Für praktische Zwecke und für ein optimales Betriebsverhalten von en in Echt-Zeit-empfindlichen Anwendungen können die meisten, wenn nicht sämtliche, der häufig verwendeten Analyseprimitivfunktionen als eingebaute High-Level-Analyseprimitivfunktionen 120 in dem allgemeinen Analysesystem vorgesehen werden. Kaum benutzte oder kunden-spezifische Analyseprimitivfunktionen können dann durch benutzer-definierte Analysemodule 130 bereitgestellt werden, die aus den eingebauten Low-Level-Analyseprimitivfunktionen 120 in der Weise erzeugt werden, die in den Fig. 14-16 dargestellt ist. Im Anhang C ist ein Satz von Low-Level-Analyseprimitivfunktionen 120 beschrieben. Dieser Satz umfasst eine Anzahl von Beispielen von verschiedenen Typen von Low-Level-Analyseprimitivfunktionen 120, die verwendet werden können, um die gewünschte Erweiterungsfähigkeit bereitzustellen. Wie Durchschnittsfachleute erkennen, können

jedoch viele andere Low-Level-Analyseprimitivfunktionen 120 verwendet werden.

Zusätzlich zum Bilden von anwendungs-spezifischen oder benutzer-spezifischen Analysen können die Low-Level-Analyseprimitive 120 verwendet werden, um eine neue Funktionalität in das allgemeine Analysesystem einzuführen, nachdem eine bestimmte Version freigegeben worden ist. Die neue Funktionalität kann z.B. (i) das allgemeine Analysesystem an die bestimmten Anforderungen, z.B. einer Statistik-Abtastung, eines spezifischen Kunden, z.B. einer Bell Operating Company, anpassen oder (ii) einen spezifischen Analysetyp zum Verwenden einer neuen Analysemodulfamilie anpassen, z.B. ein Vergleich eines Aufenthaltsorts eines mobilen Teilnehmers mit einem festen Aufenthaltsort eines Teilnehmers, oder (iii) allgemeine Erweiterungen an der existierenden Funktionalität bereitzustellen. Die neue Funktionalität kann danach über eingebaute Analyseprimitive 120 für ein besseres Betriebsverhalten (Ausführungsgeschwindigkeit) in nachfolgenden Versionen des allgemeinen Analysesystems bereitgestellt werden, (d.h. die High-Level-Primitivfunktionen können durch einen direkt kompilierten Code anstelle durch den gefädelten Code, der sich indirekt auf die Low-Level-Primitive bezieht, bereitgestellt werden.

Kompilierungszeit- und Ausführungszeit-Unterschiede zwischen Analyseprimitiven und Analysemodulen

Im Prinzip behandelt das allgemeine Analysesystem benutzer-definierte Analysemodule 130 genauso wie eingebaut Analyseprimitive 120. Es gibt jedoch bestimmte Unterschiede in ihrem Kompilierungszeit- und Ausführungszeit-Verhalten. Insbesondere werden die externen Namen der Analyseprimitive 120 in dem Analyseprimitivfunktions-Wörterbuch 104 gefunden werden, während die externen Namen der Analysemodule 130 in dem Analysemodul-Wörterbuch einer bestimmten Analyse gefunden werden, z.B. in dem Analysemodul-Wörterbuch 138 in den

Routing-Analyse-Steuerdaten 136. Im Gegensatz dazu kann das Primitivfunktions-Token 162 (oder irgendwelche anderen Funktionstoken) in irgendeinem Analyseprimitiv 120 oder einem Analysemodul 130 entweder die Analyseprimitivfunktion-Tokentabelle 106 oder die Analysemodul-Tokentabelle 146 indizieren (siehe z.B. die Fig. 15, wo die Funktionstoken in dem High-Level-Analysemodul 130b sowohl auf das Low-Level-Analysemodul 130a als auch auf mehrere Low-Level-Analyseprimitive 120a, 120b, 120e, 120f, 120g, 120h, 120i, 120j, 120k, 120l und 120m verweisen).

Während des Kompilierungszustands, wenn der Betreiber einen Befehl durch die Betreiberschnittstelle 200 zum Bilden eines neuen Analysemoduls 130 innerhalb einer bestimmten Analyse eingibt, z.B. die Routing-Analyse-Steuerdaten 136, kann die Eingangsdefinition des neuen Analysemoduls 130 die externen Namen entweder von eingebauten Analyseprimitiven 120 oder vorher definierten Analysemodulen 130 enthalten. Zum Beispiel kann die Definition eines neuen High-Level-Analysemoduls 130 die Namen von Low-Level-Analyseprimitiven 120 und Low-Level-Analysemodulen 130 enthalten. In ähnlicher Weise kann die Definition eines neuen Steuerebenen-Analysemoduls 130 ein definierendes Wort verwenden, das entweder ein High-Level-Analyseprimitiv 120 oder ein High-Level-Analysemodul 130 ist, und kann die Namen von mehreren vorher definierten Steuerebenen-Analysemodulen 130 enthalten. Demzufolge müssen unter Umständen sowohl das Analyseprimitivfunktions-Wörterbuch 104 als auch das Analysemodul-Wörterbuch 138 durchsucht werden, um jedes Analyseprimitiv 120 oder Analysemodul 130 zu finden, das in der Eingabetextkette genannt ist.

Während des Ausführungszustands, nachdem das neue Analysemodul 130 gebildet worden und zu den Routing-Analysesteuerdaten 136 hinzugefügt worden ist, wenn das Anrufsteuerprogramm 56 eine Aufforderung zum Ausführen der Routing-Analyse-Steuerdaten 136 sendet, dann kann das Funktionstoken 162 (und irgendwelche anderen Funktionstoken)

in dem neuen Analysemodul 130 entweder auf ein eingebautes Analyseprimitiv 120 oder auf eine benutzer-definiertes Analysemodul 130 verweisen. Demzufolge müssen unter Umständen sowohl die Analyseprimitivfunktions-Tokentabelle 106 als auch die Analysemodul-Tokentabelle 146 durchsucht werden, um das Primitivfunktionstoken 162 (und irgendwelche anderen Tokens) in dem neuen Analysemodul 130 in einen Zeiger zu dem Analyseprimitiv 120 oder dem Analysemodul 130 zu übersetzen, das früher von dem Betreiber verwendet wurde, um das neue Analysemodul 130 zu erzeugen.

Mehrere Suchstrategien können entwickelt werden, um die Analyseprimitive 120 und die Analysemodule 130 während einer Kompilierung oder Ausführung zu lokalisieren. Zum Beispiel kann bei der Kompilierung eines Steuerebenen-Analysemoduls 130 für eine Routing-Analyse die Suche nach der High-Level-Primitivfunktion (definierendes Wort) zunächst mit den benutzer-definierten Analysemodulen 130 gefolgt von den eingebauten Analyseprimitivfunktionen 120 starten, d.h. die Strategie kann sein, das Analysemodul-Wörterbuch 120 in den Routing-Analyse-Steuerdaten 136 nach dem Namen der von dem Betreiber eingegebenen Primitivfunktion zu durchsuchen und wenn diese Suche keine Lokalisierung des Namens ergibt, wird eine Suche in dem Analyseprimitivfunktions-Wörterbuch 104 begonnen (dieser Suchbefehl ist empfohlen, da der Benutzer z.B. wählen kann, ein Analysemodul 130 zu erzeugen, welches eine spezielle Version eines Analyseprimitivs 120 ist, und den Namen des eingebauten Analyseprimitivs 120 auch für das benutzer-definierte Analysemodul 130 zu verwenden). Alle Standardsuch-Beschleunigungstechniken, einschliesslich Hash-Tabellen, können beim Ausführen dieser Wörterbuch-Durchsuchungen verwendet werden.

Für ein effizientes Durchsuchen während einer Ausführung (das zeitempfindlicher als eine Kompilierung ist) und zum Vermeiden einer Suche sowohl in der Analyseprimitiv-Funktionstokentabelle 106 als auch der Analysemodul-

Token-tabelle 146 kann das allgemeine Analysesystem jedoch mit der Fähigkeit versehen werden, zwischen dem Primitivfunktionstoken und den Analysemodultoken zu unterscheiden (die auf die eingebauten Analyseprimitive 120 bzw. die benutzer-definierten Analysemodule 130 verweisen). Während viele Techniken zum Differenzieren zwischen den Primitivfunktions-Token und den Analysemodul-Token innerhalb der bevorzugten Ausführungsform der vorliegenden Form der Erfindung angewendet werden können, kann eine relativ simple Technik darin bestehen, zwei unterschiedliche Zahlbereiche, z.B. positive und negative Zahlen, für die Primitivfunktionstoken bzw. die Analysemodultoken zu verwenden. Wenn positive und negative Zahlen verwendet werden, kann dem höchstwertigen Bit in dem Primitivfunktionstoken z.B. der Wert 0 zugewiesen werden und dem höchstwertigen Bit in den Analysemodul-Token kann der Wert 1 (oder umgekehrt) zugewiesen werden. In dieser Weise kann das allgemeine Analysesystem leicht bestimmen, welche der zwei Token-tabellen 106 und 146 während einer Ausführung zu durchsuchen sind.

Es sei darauf hingewiesen, dass in der bevorzugten Ausführungsform der vorliegenden Erfindung eine getrenntes Wörterbuch und eine getrennte Token-tabelle für die Analyseprimitive 120 (Fig. 10) einerseits und für die Analysemodule 130 in jeder Analysesteuerdatenstruktur (Fig. 11) andererseits vorgesehen sind. In einer alternativen Ausführungsform können die externen Namen und Token sowohl der Analyseprimitive 120 als auch der Analysemodule 130 von sämtlichen verschiedenen Analysen in dem gleichen Wörterbuch bzw. der gleichen Token-tabelle gespeichert werden. Ein möglicher Vorteil einer Trennung der Analysemodule 130 von den Analyseprimitiven 120 und dann einer Trennung der Analysemodule 130 für eine Analyse von den Analysemodulen 130 für eine andere Analyse besteht jedoch darin, dass der Benutzer des allgemeinen Analysesystems verschiedene Analysemodule für jede unterschiedliche Analyse, anstelle dass ein grosser Satz für alle Analysen vorgesehen ist, bilden kann,

d.h. der Benutzer erweitert den Satz von Analysemodulen für jede Analyse in einer modularen Weise. Ein anderer möglicher Vorteil einer Trennung der eingebauten Analyseprimitive 120 von den benutzer-definierten Analysemodulen 130 in der bevorzugten Ausführungsform besteht darin, das die Analyseprimitive 120 an einer Stelle in einem Speicher gespeichert werden können, dem für Sicherheitszwecke ein Schreibschutz (ein Nur-Lese-Zugriff) gegeben wird, während die Analysemodule 130 an einer anderen Stelle in dem Speicher gespeichert werden können, auf die der Benutzer einen Lese- und Schreibzugriff aufweist.

Betreiberschnittstelle

Wie sich von Durchschnittsfachleuten erkennen lässt, können mehrere Kopplungstechniken im Zusammenhang mit der Betreiberschnittstelle 200 implementiert werden. In der bevorzugten Ausführungsform der vorliegenden Erfindung wird ein Telecommunication Management Network (TMN) Rahmenwerk verwendet und die Betreiberschnittstelle 200 ist deshalb eine geschichtete Schnittstelle gemäss dem Open System Interconnection (OSI) Modell. Siehe zum Beispiel ANSI T1-210-1989 American National Standard for Telecommunications-Operations, Administration, Maintenance and Provisioning (OAM & P)- Principle of Functions Architectures and Protocols for Interfaces between Operation Systems and Network Elements.

In der oberen Schicht des OSI Modells wird das Protokoll der Common Management Information Service Elements (CMISE) verwendet. (CMISE) stellt einen Satz von generischen Befehlen, beispielsweise Set (Einstellen), Get (Holen), Delete (Löschen), Modify (Modifizieren) etc. bereit, die auf die verwalteten Objekte angewendet werden können. In dem allgemeinen Analysesystem der vorliegenden Erfindung umfassen die verwalteten Objekte die Analysemodule 130 in einem Analysesteuerdatenwert, z.B. den Routing-Analyse-Steuerdaten 136. Die Information, die durch CMISE weitergeleitet wird, um

spezifische Werte für die verwalteten Objekt zu spezifizieren, ist kontext-abhängig, d.h. sie hängt von dem Typ des verwalteten Objekts ab. Überdies könnte diese kontext-abhängige Information genauso durch andere Einrichtungen, wie einer Mensch-Maschine-Schnittstelle weitergeleitet werden, z.B. einem Terminal, welches direkt mit dem allgemeinen Analysesystem verbunden ist, das die Mensch-Maschine-Sprache (MML) definiert von CCITT oder eine spezielle Syntax, die für das allgemeine Analysesystem entworfen ist, verwendet.

Die verschiedenen möglichen Implementierungen der Betreiberschnittstelle 200 können verschiedene Techniken verwenden, um die Analysemodule 130 zu identifizieren, können aber eine gemeinsame Technik verwenden, um deren internen Aufbau auszudrücken. Unter Verwendung der normalen Einrichtung einer Threaded Interpretative Language (gefädelten Interpretationssprache TIL) kann die Struktur und der Wert eines Analysemoduls 130 in einem TIL Quellenformat ausgedrückt werden, d.h. einer Textkette (Wörter und Zahlen), die die Definition des Analysemoduls enthalten. In der Praxis kann das allgemeine Analysesystem unter Umständen mehr als eine Schnittstelle gleichzeitig unterstützen und die Betreiberschnittstelle 200 kann dann so betrachtet werden, dass sie eine externe Systemschnittstelle bildet, die mehreren externen Schnittstellen gemeinsam ist. Diese Situation ist allgemein in Fig. 17 dargestellt.

Die in Fig. 17 gezeigte Betreiberschnittstelle 200 kann z.B. als ein Satz von Unterprogrammen gebildet werden, die einen Satz von generischen Operationen bereitstellen, die verwendet werden, um neue Analysemodule zu definieren, um nicht verwendete Analysemodule zu löschen oder zu ersetzen, um die Verwendung der aktiven und passiven Analysemodul-Token-tabelle innerhalb irgendwelcher Analysesteuerdaten etc. zu löschen. Der benötigte Eingang zu einem Unterprogramm, das z.B. ein neues Analysemodul in den Analysesteuerdaten einfügt, kann umfassen: (i) eine Identifikation der Analysesteuerdaten, d.h.

den externen Namen der Analyse, (ii) eine Identifikation des Analysemoduls, d.h. den externen Namen des Analysemoduls, und (iii) die Definition der Inhalte des Analysemoduls, das in einem TIL Quellenformat in der voranstehend beschriebenen Weise (Kompilierungsbeispiele) angeordnet ist.

In Fig. 17 passt eine lokaler Schnittstellenadapter 240 die interne Systemschnittstelle 200 an eine externe Systemschnittstelle 243 zu einem nächsten Terminal 245 an, an dem eine Betreiber die Analysesteuerdaten für verschiedene Analysen, z.B. die Routing-Analyse-Steuerdaten 136, hinzufügen und modifizieren kann. Die externe Schnittstelle 243 kann z.B. als eine textgestützte Schnittstelle gebildet werden, die konsistent mit den vorher für den Kompilierungszustand der vorliegenden Erfindung angegebenen Beispielen ist. Der lokale Schnittstellenadapter 240 übersetzt den Textstrom von dem Betreiber des Terminals 245 in Sequenzen von Unterprogramm-Aufrufen in der internen Systemschnittstelle 200. Ein CMISE-gestützter Schnittstellenadapter 241 wird auch zum Kommunizieren mit einem externen Betriebsunterstützungssystem 242 durch eine Schnittstelle 248 verwendet. Das Betriebsunterstützungssystem 242 ist ein Unterstützungssystem, das für die Verwaltung von mehreren Telekommunikationssystemen, z.B. Vermittlungen in einem Telekommunikationsnetz verwendet wird. Das Betriebsunterstützungssystem ist mit einem Terminal 247 durch eine Schnittstelle 246 verbunden, die zum Beispiel ein Grafikeditor sein kann.

Es gibt mehrere verschiedenen Typen von Schnittstellen in der heutigen Verwendung zwischen Betriebsunterstützungssystemen und Telekommunikationssystemen. Für Zwecke der vorliegenden Erfindung wird angenommen, dass die Schnittstelle 248 zwischen dem Betriebsunterstützungssystem 242 und den Telekommunikationssystemen, die das allgemeine Analysesystem der vorliegenden Erfindung verwenden, auf dem Rahmenwerk gestützt ist, das in ANSI T1-210 1989 angeführt ist. Diese

Schnittstelle kann auf einer direkten Datenstrecke zwischen dem Betriebsunterstützungssystem 242 und den Telekommunikationssystemen, die das allgemeine Analysesystem verwenden oder durch ein Datenkommunikationsnetz unter Verwendung der CCITT Empfehlung X.25 verbunden werden sollen. Die Schnittstelle 248 umfasst Protokolle auf allen sieben Schichten des OIS-Modells. In der obersten Schicht wird die CMISE Anwendungsdienst-Einheit (ISO 9595-2, 9596-2) verwendet.

Der CMISE-gestützte Schnittstellenadapter 241 verbindet die Schnittstelle 200 mit dem Betriebsunterstützungssystem 242 für die Telekommunikationssysteme, die das allgemeine Analysesystem verwenden und übersetzt die empfangenen CMISE Aktionen in Sequenzen von Unterprogramm-Aufrufen in der internen Systemschnittstelle 200. Die CMISE Vorgänge umfassen generische Befehle wie Set (Einstellen), Get (Holen), Create (Erzeugen), Delete (Löschen) etc. Dies Operationen enthalten Parameter oder Attribute, die die Objekte identifizieren, an die die Aktion gerichtet ist. Zum Beispiel wird eine Aktion, die ein neues Analysemodul in die Analysesteuerdaten einfügt, Parameter enthalten, die die Analyse (entsprechend dem externen Namen der Analyse), eine Identifikation von Analysemodulen (entsprechend zu dem externen Namen des Analysemoduls) und die Definition der Inhalte des bestimmten Analysemoduls enthalten werden. In dem angenommenen Rahmenwerk wird der CMISE Betrieb unter Verwendung der ASN.1 Basic Encoding Rules definiert in CCITTrec.X.209 kodiert. Die Inhalte des Analysemoduls können noch unter Verwendung eines eingebetteten TIL Quellenformats definiert werden, jedoch in einem Parameter innerhalb des kodierten CMISE Betriebs.

Die früher beschriebenen Beispiele der Verwendung des Eingangs zu dem allgemeinen Analysesystem, um (i) aus einem High-Level-Analysemodul ein neues Steuerebenen-Analysemodul innerhalb einer Analyse zu erzeugen, (ii) ein neues Low-Level-Analysemodul zu erzeugen, welches später in der Definition von anderen Analysemodulen enthalten sein kann, (iii) ein neues

High-Level-Analysenmodul zu erzeugen, welches später verwendet werden kann, um neue Steuerebenen-Analysenmodule innerhalb einer Analyse zu erzeugen; und (iv) aus High-Level-Analysenmodulen neue Steuerebenen-Analysenmodule zusammenzustellen, sind alle auch für den Fall gültig, dass der Eingang zu dem allgemeinen Analysesystem gerade den CMISE-gestützten Schnittstellenadapter 241 verwendet, obwohl die Codierung der verschiedenen Eingangsparameter, die von dem allgemeinen Analysesystem angefordert werden, etwas anders sein werden.

Mit dem TIL-Merkmal wird deshalb sogar eine generische Betriebsschnittstelle, z.B. CMISE gemäß der vorliegenden Erfindung erweiterbar. Neue Funktionen können hinzugefügt werden und die generische Schnittstelle kann komplexer gebildet werden, ohne die Schnittstelle tatsächlich zu ändern. Die erweiterbare Schnittstelle kann verwendet werden, um die neue Analyseprimitivfunktionen der eingebauten Analyseprimitivfunktionen, die in einzigartiger Weise für eine bestimmte Analyse benötigt werden, hinzuzufügen. Während die Benutzer-definierten Analyseprimitivfunktionen einen Einfluss auf die Echtzeitverarbeitungskapazität des allgemeinen Analysesystems aufweisen werden, können diejenigen, die sich als relativ nützlich erweisen, immer in der nächsten Version des allgemeinen Analysesystems mittels Hardware codiert werden, und zwar für ein besseres Betriebsverhalten, ohne irgendwelche der Definitionen in den dann existierenden Analysen zu modifizieren.

Wesentliche Punkte des allgemeinen Analysesystems

Um einige Merkmale der vorliegenden Erfindung, die nun deutlich angesichts der voranstehenden Diskussion verstanden werden kann, ins Gedächtnis zurückzuholen, kann eine verallgemeinerte Struktur von Analysesteuerdaten, d.h. die Analysesteuerdaten 42 genauso wie ein Programm in einer Datenbank gespeichert und mit einem verallgemeinerten

Interpretierer, d.h. dem allgemeinen Analyseprogramm 40, interpretiert werden. Jede spezifische Analyse, z.B. eine Routing-Analyse, kann durch eine Anzahl von Analysefunktionen dargestellt werden, d.h. die (Steuerebenen-)Analysemodule 130, die als Daten in den Analysesteuerdaten 42 gespeichert sind. Wenigstens einige der Analysemodule 130 arbeiten mit einem bestimmten Parameter (mit bestimmten Parametern), um eine nützliche Analyse zu erzeugen und diese und andere Analysemodule 130 können kombiniert und rekombiniert werden, um eine vollständigere Analyse zu erzeugen. Eine standardisierte Struktur für jedes der (Steuerebenen-)Analysemodule 130 umfasst einen Zeiger auch zu einem spezifischen Interpretierer-Unterprogramm, d.h. eine Analyseprimitivfunktion 120, die einen spezifischen Typ von (Steuerebenen-)Analysemodule 130 in grob der gleichen Weise in jeder Analyse analysiert, und einen Zeiger auf ein anderes (Steuerebenen-)Analysemodul 130, das der Analyse folgt. Demzufolge kann eine Analyse durch Ändern der Zeiger zwischen den (Steuerebenen-)Analysemodulen 130 leicht geändert werden.

Wenn eine Analyse erfordert, dass eine neue Funktion zu der Analyse hinzugefügt wird oder ein neuer Parameter analysiert wird, können ein oder mehrere neue Analysemodule 130 innerhalb der Analysesteuerdaten 42 gebildet werden, um die neue Funktionalität bereitzustellen und mit dem neuen Parameter zu arbeiten. Das allgemeine Analysesystem der vorliegenden Erfindung ist deshalb ausdehnbar und erweiterbar, von dem Standpunkt von sowohl neuen Parametern als auch neuen Typen von Analysen. Wegen der verallgemeinerten Struktur der Steuerdaten 42 kann die Betreiberschnittstelle ferner auch als ein Programm, welches leicht mit verschiedenen Betreiberunterstützungssystemen gekoppelt wird, verallgemeinert und konfiguriert werden. Eine High-Level-Betreiberschnittstelle kann fortgeschrittene Werkzeuge bereitstellen, die den Betreiber beim Erzeugen von neuen Analysemodulen 130 innerhalb der Analysesteuerdaten 42 unterstützen.

In dieser Weise vermeidet der Betreiber das allgemeine Analysesystem der vorliegenden Erfindung nicht nur die großen Kosten zum Erstellen der Daten für jede Analyse in dem spezifischen Format, das von herkömmlichen Analysesystemen benötigt wird, sondern schneidet die Analyse auch zu, so dass sie unvorhergesehene Anforderungen einer besonders industriellen Umgebung standhält, z.B. einer Telefonvermittlung, um nunmehr irgendwelche Modifikationen an dem allgemeinen Analyseprogramm 40 durchzuführen, was ansonsten gemäß der herkömmlichen Analysesysteme erforderlich sein würde. Dies wird mit dem neuartigen Konzept der vorliegenden Erfindung möglich, da es sich um die Daten handelt (Analysesteuerdaten 42), die anstelle des Programms unter Verwendung der Daten das Programm (allgemeines Analyseprogramm 40) verwenden, d.h. jedes (Steuer-ebenen-)Analysemodul 130 weist selbst aus, welches Unterprogramm (Analyseprimitivfunktion 120) beim Interpretieren der Daten, die in dem Analysemodul 130 enthalten sind, verwendet werden soll, anstelle des Unterprogramms, welches ausweist, welche Daten interpretiert werden sollen. Gemäß der vorliegenden Erfindung können neue Funktionen und neue Parameter, die nicht vorhergesagt und für eine bestimmte Version des allgemeinen Analyseprogramms 40 berücksichtigt wurden, leicht angepasst werden, indem neue Analysemodule 130 (einschließlich von Analysemodulen 130, die wie Analyseprimitivfunktionen 120 arbeiten) in den Analysesteuerdaten 42 definiert werden.

Die voranstehend beschriebene ausführliche Beschreibung zeigt nur bestimmte besondere Ausführungsformen der vorliegenden Erfindung. Jedoch werden Durchschnittsfachleute erkennen, dass zahlreiche Modifikationen und Variationen möglich sind, ohne im wesentlichen von dem Umfang der vorliegenden Erfindung abzuweichen. Demzufolge sei deutlich darauf hingewiesen, dass die Form der hier beschriebenen Erfindung nur beispielhaft ist, und dies nicht als Einschränkung des Umfangs der

Erfindung, wie in den folgenden Ansprüchen definiert,
vorgesehen ist.

PATENTANSPRÜCHE

1. Allgemeines Analysesystem, umfassend:

Analysesteuerdaten (42) umfassend eine Vielzahl von Analysemodulen (70, 130); und

ein allgemeines Analyseprogramm (40) zum Ausführen wenigstens einer Analyse für einen Satz von Daten, wobei das allgemeine Analyseprogramm (40) eine Vielzahl von Analyseprimitiven (78, 120) umfasst, wobei die wenigstens eine Analyse aus einem Satz der Vielzahl von Analysemodulen (70, 130) konstruiert ist, wobei die Ausführung durch Verwendung wenigstens eines der Analyseprimitiven (78, 120) ausgeführt wird.
2. Allgemeines Analysesystem nach Anspruch 1, dadurch gekennzeichnet, dass das allgemeine Analyseprogramm (40) ferner eine Einrichtung (102, 114) zum Hinzufügen von neuen Analysemodulen umfasst.
3. Allgemeines Analysesystem nach Anspruch 1, dadurch gekennzeichnet, dass jedes der Analyseprimitive (78, 120) einen bestimmten Typ von Analysemodulen (70, 130) interpretiert.
4. Allgemeines Analysesystem nach Anspruch 3, dadurch gekennzeichnet, dass jedes der Analysemodule (70, 130) umfasst:

eine Anzeige (72, 162) des Analyseprimitivs (78, 120), das seinen Typ interpretiert; und

Steuerdaten (60), die einen Teil der wenigstens einen Analyse steuern.

5. Allgemeines Analysesystem nach Anspruch 1,
dadurch gekennzeichnet, dass
die Analysesteuerdaten (42) und das allgemeine Analyseprogramm (40) in einer Telefonvermittlungsstelle (12, 20, 22) angeordnet ist und der Datensatz umfasst:

Anrufsteuerdaten (54) mit einer Vielzahl von Anrufaufzeichnungen (116), jeweils eine für jeden Telefonanruf durch die Vermittlung.
6. Allgemeines Analysesystem nach Anspruch 5,
dadurch gekennzeichnet, dass
die Analysemodule (70, 130) in eine Vielzahl von Analysestrukturen gruppiert sind, wobei jede eine andere Analyse steuert.
7. Allgemeines Analysesystem nach Anspruch 6,
dadurch gekennzeichnet, dass
die Analysestrukturen eine Wegleitungsanalyse-Struktur (136) und eine Zielstellenanalyse-Struktur (134) umfassen.
8. Allgemeines Analysesystem nach Anspruch 6,
dadurch gekennzeichnet, dass
jede Anrufaufzeichnung (116) eine Vielzahl von Anrufparameterwerten speichert, für wenigstens eine der folgenden:

Teilnehmerdaten (50) mit Daten über das Dienstprofil eines einleitenden Teilnehmers;

eine Teilnehmereingabe (58) mit den Ziffern, die von dem einleitenden Teilnehmer gewählt werden;

ein Ergebnis (Ergebnisse) (48) einer Analyse, die von dem allgemeinen Analyseprogramm (40) unter der Steuerung einer der Analysestrukturen ausgeführt wird, wobei die Ergebnisse in die Anrufaufzeichnung (116) von dem allgemeinen Analyseprogramm (40) geschrieben werden; und

andere Daten, die in die Anrufaufzeichnung von einem Anrufsteuerprogramm (56) geschrieben werden, das dem allgemeinen Analyseprogramm (40) befiehlt, andere Analysen auszuführen.

9. Allgemeines Analysesystem nach Anspruch 8, dadurch gekennzeichnet, dass das Ergebnis (die Ergebnisse) (48) von dem allgemeinen Analyseprogramm (40) verwendet wird (werden), um eine andere Analyse auszuführen.

10. Allgemeines Analysesystem nach Anspruch 8, dadurch gekennzeichnet, dass das Ergebnis (die Ergebnisse) (48) von dem Anrufsteuerprogramm (56) verwendet werden, um eine Anrufverarbeitung zu steuern.

11. Allgemeines Analysesystem nach Anspruch 1, dadurch gekennzeichnet, dass:

das allgemeine Analyseprogramm (40) eine Vielzahl von Low-Level-Analyseprimitiven (120a-m) und eine Vielzahl von High-Level-Analyseprimitiven (120) umfasst, wobei jedes der Low-Level-Analyseprimitive (120a-m) ein Unterprogramm umfasst, das eine bestimmte Funktion ausführt, wobei jedes der High-Level-Analyseprimitive (120) eine Bildungsprozedur (114) und eine Do-Prozedur (112) aufweist; und

die Analysesteuerdaten (42) eine Vielzahl von Low-Level-Analysemodulen (130a), eine Vielzahl von High-Level-

Analysemodulen (120b) und eine Vielzahl von Steuerebenen-Analysemodule (130, 130c) umfassen, wobei jedes der Low-Level-Analysemodule (130a) aus wenigstens einem der Low-Level-Analyseprimitive (120a-m) kompiliert wird, wobei jedes der High-Level-Analysemodule (130b) eine Bildungsprozedur und eine Do-Prozedur aufweist, wobei jedes der Steuerebenen-Analysemodule (130, 130c) mit der Bildungsprozedur kompiliert wird und mit der Do-Prozedur von einem der High-Level-Analyseprimitive (120) oder einem der High-Level-Analysemodule (130b) interpretiert wird.

12. Allgemeines Analysesystem nach Anspruch 11, **dadurch gekennzeichnet**, dass die Low-Level- und High-Level-Analyseprimitive (120, 120a-m) eingebaut sind und die Low-Level- und High-Level- und Steuerebenen-Analysemodule (130, 130a-c) benutzerdefiniert sind.
13. Allgemeines Analysesystem nach Anspruch 11, **dadurch gekennzeichnet**, dass die Steuerebenen-Analysemodule (130, 130c) einen Funktionsanzeiger (162), der einen Bezug zu dem interpretierenden High-Level-Analyseprimitive (120) oder einem High-Level-Analysemodul (130b) herstellt, und ein Steuerfeld (60) umfasst, das Steuerdaten zum Steuern eines Teils einer Analyse enthält.
14. Allgemeines Analysesystem nach Anspruch 13, **dadurch gekennzeichnet**, dass jedes von wenigstens einigen der Steuerebenen-Analysemodule (130, 130c) ferner eine Parameteridentifikation (74) umfasst, die einen Bezug zu einem Parameter (46) herstellt, der von dem Steuerebenen-Analysemodul (130, 130c), auf das Bezug genommen wird, analysiert werden soll.

15. Verfahren zum Steuern eines industriellen Prozesses, umfassend die folgenden Schritte:

Speichern einer Vielzahl von Parametern (46) für den industriellen Prozess, wobei jeder einzelne der Vielzahl von Parametern (46) umfasst:

einen Eingang, der von dem industriellen Prozess erzeugt wird;

Daten, die die Charakteristiken des industriellen Prozesses definieren; und

ein Ergebnis (Ergebnisse) von früher ausgeführten Analysen;

Ausführen einer Vielzahl von Analysemodulen (70, 103), wobei jedes einen Teil einer Analyse der Parameter (46) steuert und wenigstens eines davon ein Ergebnis von der Analyse erzeugt; und

Steuern des industriellen Prozesses auf Grundlage des Ergebnisses.

16. Verfahren nach Anspruch 15, dadurch gekennzeichnet, dass das Ergebnis ein Zwischenergebnis aus der Analyse ist.

17. Verfahren nach Anspruch 15, dadurch gekennzeichnet, dass das Ergebnis ein Endergebnis aus der Analyse ist.

18. Verfahren nach Anspruch 15, dadurch gekennzeichnet, dass der industrielle Prozess ein Prozess für die Verbindung von Telefonanrufen ist und die Parameter Anruf-bezogene Daten sind.

19. Verfahren nach Anspruch 15,
dadurch gekennzeichnet, dass
jedes der Analysemodule (70, 130) umfasst:
- einen Funktionsanzeiger (72, 162), der eine Funktion (78, 120) anzeigt, die ausgeführt werden soll, wenn das Analysemodul, auf das Bezug genommen wird, ausgeführt wird; und
- ein Steuerfeld (60), das Steuerdaten enthält, die von der Funktion (78, 120) interpretiert werden sollen.
20. Verfahren nach Anspruch 19,
dadurch gekennzeichnet, dass das Steuerfeld (60)
wenigstens einen Modulanzeiger (80) umfasst, der das
nächste Analysemodul anzeigt, welches ausgeführt werden
soll.
21. Verfahren nach Anspruch 20,
dadurch gekennzeichnet, dass das Steuerfeld (60)
wenigstens eine Parameteridentifikation (74) umfasst, die
die Stelle eines Parameters (46) in einem Speicher (116)
identifiziert.
22. Verfahren nach Anspruch 21,
dadurch gekennzeichnet, dass die Funktion (78, 120) den
Parameterwert analysiert, der in der Speicherstelle
gespeichert ist.
23. Verfahren nach Anspruch 21,
dadurch gekennzeichnet, dass das Steuerfeld (60) ferner
wenigstens einen vorgegebenen Wert umfasst und die
Funktion (78, 120) den vorgegebenen Wert in die
Speicherstelle schreibt.

24. Verfahren zum Kompilieren und Ausführen wenigstens einer Analyse, die zum Steuern eines industriellen Prozesses verwendet wird, umfassend die folgenden Schritte:

Bereitstellen einer Vielzahl von High-Level-Analyseprimitiven (78, 120), die jeweils eine Bildungsprozedur (114) und eine Do-Prozedur (112) umfassen;

Kompilieren einer Vielzahl von Steuerebenen-Analysemodulen (70, 130) unter Verwendung der Bildungsprozedur (114) von wenigstens einem der High-Level-Analyseprimitive (78, 120); wobei die Steuerebenen-Analysemodule (70, 130) eine Baumstruktur bilden, die die wenigstens eine Analyse darstellt; und

Ausführen jedes Steuerebenen-Analysemoduls (70, 130) in der Baumstruktur unter Verwendung der Do-Prozedur (112) des High-Level-Analyseprimitivs (78, 120), das verwendet wurde, um das Steuerebenen-Analysemodul (70, 130) zu kompilieren.

25. Verfahren nach Anspruch 24, ferner umfassend die folgenden Schritte:

Bereitstellen einer Vielzahl von Low-Level-Analyseprimitive (120a-m);

Kompilieren einer Vielzahl von Low-Level-Analysemodulen (130a) unter Verwendung der Low-Level-Analyseprimitiven (120a-m);

Kompilieren einer Vielzahl von High-Level-Analysemodulen (130b) unter Verwendung der Low-Level-Analysemodule (130a), wobei jedes der High-Level-Analysemodule (130b) eine Bildungsprozedur und eine Do-Prozedur umfasst;

Kompilieren einer Vielzahl von neuen Steuerebenen-Analysemodulen (130c) unter Verwendung der Bildungsprozedur von wenigstens einem der High-Level-Analysemodule (130b), wobei die neuen Steuerebenen-Analysemodule (130c) zu der Baumstruktur hinzugefügt werden; und

Ausführen jedes neuen Steuerebenen-Analysemoduls (130c) in der Baumstruktur unter Verwendung der Do-Prozedur des High-Level-Analysemoduls (130b), welches verwendet wurde, um das neue Steuerebenen-Analysemodul (130c) zu kompilieren.

26. Verfahren nach Anspruch 25,
dadurch gekennzeichnet, dass
die High-Level-Analyseprimitive (78, 120) und die Low-Level-Analyseprimitive (120a-m) eingebaute Funktionen sind.
27. Verfahren nach Anspruch 25,
dadurch gekennzeichnet, dass
die High-Level-Analysemodule (130b) und die Low-Level-Analysemodule (130a) benutzerdefinierte Funktionen sind.
28. Allgemeines Analysesystem, umfassend:

eine Vielzahl von Analysesteuerdatenstrukturen (134, 136) für eine Vielzahl von Analysen, die jeweils einen externen Namen aufweisen, wobei jede der Analysesteuerdatenstrukturen (134, 136) umfasst:

eine Vielzahl von Analysemodulen (70, 130), die jeweils einen externen Namen aufweisen und wobei wenigstens einige von diesen ausgeführt werden, wenn eine Analyse ausgeführt wird;

ein Analysemodul-Wörterbuch (138), welches die Namen der Analysemodule (70, 130) in entsprechende Modultoken umwandelt; und

wenigstens eine Analysemodul-Tokentabelle (146, 148), die die Modultoken in entsprechende Zeiger auf die Analysemodule (70, 130) übersetzt;

ein Analysesteuerdaten-Wörterbuch (132), das die Namen der Analysen in entsprechende Zeiger zu den Analysesteuerdatenstrukturen (134, 136) übersetzt;

eine Vielzahl von Analyseprimitivfunktionen (78, 120), die jeweils einen externen Namen aufweisen und wobei wenigstens einige von diesen bestimmte Typen der Analysemodule (70, 130) kompilieren und interpretieren;

ein Analyseprimitivfunktions-Wörterbuch (140), das die Namen der Analyseprimitivfunktionen (78, 120) in entsprechende Primitivfunktionstoken übersetzt;

eine Analyseprimitivfunktions-Tokentabelle (106), die die Primitivfunktionstoken in entsprechende Zeiger zu den Analyseprimitivfunktionen (78, 120) übersetzt;

eine Analyseinterpretierprozedur (108), die die Ausführung einer gewählten Analyse im Ansprechen auf den Empfang von Eingangsdaten (44) initiiert, die den Namen der gewählten Analyse und einen Parameter-Aufzeichnungszeiger umfassen, der auf eine Aufzeichnung (116) weist, die einen Satz von Parametern (46) enthält, der analysiert werden soll;

einen Analyseausführungs-Eingangsdaten-Behandler (110), der den Parameteraufzeichnungszeiger speichert und ermöglicht, dass die Analyseprimitivfunktionen (78, 120)

auf die Parameter (46) während der Ausführung der gewählten Analyse zugreifen; und

eine Analyse-Bildungsprozedur (102), die die Kompilierung eines neuen Analysemoduls (neuen Analysemodulen) (70, 130) im Ansprechen auf den Empfang eines Befehls initiiert, der den Namen (die Namen) wenigstens einer Analyse in wenigstens einer der Analysesteuerdatenstrukturen (134, 136), den Namen (die Namen) des Analysemoduls (der Analysemodule) (70, 130), die kompiliert werden sollen, und den Namen (die Namen) der Analyseprimitivfunktion (der Analyseprimitivfunktionen) (78, 120), die beim Kompilieren des neuen Analysemoduls (der neuen Analysemodule) (70, 130) verwendet werden sollen, umfasst.

29. Allgemeines Analysesystem nach Anspruch 28, **dadurch gekennzeichnet**, dass jede der wenigstens einigen Analyseprimitiv-Funktionen (78, 120) eine Bildungsprozedur (114) umfasst, die zum Kompilieren eines bestimmten Typs von Analysemodulen verwendet wird, und eine Do-Prozedur (112), die zum Interpretieren des Typs von Analysemodulen (70, 130) verwendet wird, umfasst, wobei die Analyseprimitivfunktions-Tokentabelle (106) das Primitivfunktionstoken für jede der wenigsten einigen Analyseprimitivfunktionen (78, 120) in zwei Zeiger übersetzt, einen ersten Zeiger zu der Bildungsprozedur (114) und einen zweiten Zeiger zu der Do-Prozedur (112).
30. Allgemeines Analysesystem nach Anspruch 28, **dadurch gekennzeichnet**, dass die Eingangsdaten (44) von einem Anrufsteuerprogramm (56) gesendet werden, das die Verarbeitung von dem

Telefonverkehr durch ein Telekommunikationssystem steuert; und

die Parameter (46) Daten umfassen, die sich wenigstens auf einen Telefonanruf beziehen.

31. Allgemeines Analysesystem nach Anspruch 28, dadurch gekennzeichnet, dass

die Eingangsdaten (44) ferner andere Information enthalten, die für die gewählte Analyse spezifisch ist; und

die andere Information in dem Analyseausführungs-Eingangsdaten-Behandler (110) gespeichert wird.

32. Allgemeines Analysesystem nach Anspruch 31, dadurch gekennzeichnet, dass

die andere Information Zeiger zu anderen Parameteraufzeichnungen (116) enthält, die analysiert werden sollen.

33. Allgemeines Analysesystem nach Anspruch 28, dadurch gekennzeichnet, dass

der Analyseausführungs-Eingangsdaten-Behandler (110) mit einem Datenspeicher-Behandler (118) kommuniziert, der das Lesen und Schreiben von Parameterwerten von der und in die Parameteraufzeichnung (116) behandelt.

34. Allgemeines Analysesystem nach Anspruch 28, dadurch gekennzeichnet, dass jede der Analysesteuerdatenstrukturen (134, 136) umfasst:

eine Vielzahl von Analysemodul-Tokentabellen (146, 148);

einen Zeiger (144) für eine aktive Tokentabelle, der auf eine aktive der Analysemodul-Tokentabellen (146) zeigt; und

ein Analysestarttoken (142), das auf das erste Analysemodul (150), welches ausgeführt werden soll, Bezug nimmt, wenn eine Analyse ausgeführt wird, und das in einen Zeiger zu dem ersten Analysemodul (150) durch die aktiven Modultokentabelle (146) übersetzt wird.

35. Allgemeines Analysesystem nach Anspruch 34, **dadurch gekennzeichnet**, dass das Analysestarttoken (142) und der Zeiger (144) für die aktive Tokentabelle in einer Analyseausführungs-Startdaten-Struktur (140) in jeder der Analysesteuerdatenstrukturen (134, 136) angeordnet ist.
36. Allgemeines Analysesystem nach Anspruch 34, **dadurch gekennzeichnet**, dass die Analyse-Bildungsprozedur (102) einen Befehl empfängt, um den Wert des Zeigers (144) für die aktive Tokentabelle zu modifizieren.
37. Allgemeines Analysesystem nach Anspruch 28, **dadurch gekennzeichnet**, dass der Name von jeder der Analysen in sowohl einem Zeiger zu der entsprechenden Analysesteuerdatenstruktur (134, 136) als auch in ein Token zu dem ersten Analysemodul (150) übersetzt wird, das in der Analyse ausgeführt werden soll.
38. Allgemeines Analysesystem nach Anspruch 28, **dadurch gekennzeichnet**, dass der Befehl über eine Benutzerschnittstelle (200) eingegeben wird, die mit der Analysebildungsprozedur (102) kommuniziert.

39. Allgemeines Analysesystem nach Anspruch 28,
dadurch gekennzeichnet, dass
die Analysebildungsprozedur (102) einen Befehl empfängt,
um die Inhalte von wenigstens einem vorher existierenden
Analysemodul (70, 130) zu modifizieren.
40. System zum Steuern eines industriellen Prozesses,
umfassend:
- eine Einrichtung zum Speichern einer Vielzahl von
Parametern (46) für den industriellen Prozess, wobei
jeder der Parameter umfaßt:
- eine Eingabe, die von dem industriellen Prozess
erzeugt wird;
- Daten, die die Charakteristiken des industriellen
Prozesses definieren; und
- ein Ergebnis (Ergebnisse) von früher ausgeführten
Analysen;
- eine Einrichtung zum Ausführen einer Vielzahl von
Analysemodulen (70, 103), wobei jede einen Teil einer
Analyse der Parameter (46) steuert und wenigstens eine
ein Ergebnis aus der Analyse erzeugt; und
- eine Einrichtung zum Steuern des industriellen Prozesses
auf Grundlage des Ergebnisses.
41. System nach Anspruch 39,
dadurch gekennzeichnet, dass
das Ergebnis ein Zwischenergebnis aus der Analyse ist.
42. System nach Anspruch 39,
dadurch gekennzeichnet, dass
das Ergebnis ein Endergebnis aus der Analyse ist.

43. System nach Anspruch 39,
dadurch gekennzeichnet, dass
der industrielle Prozess ein Prozess für die Verbindung
von Telefonanrufen ist und die Parameter (46)
anrufbezogene Daten sind.
44. System nach Anspruch 39,
dadurch gekennzeichnet, dass
jedes der Analysemodule (70, 130) umfasst:

einen Funktionsanzeiger (72, 162), der eine Funktion (78,
120) anzeigt, die ausgeführt werden soll, wenn das
Analysemodul (70, 130), auf das Bezug genommen wird,
ausgeführt wird; und

ein Steuerfeld (60), welches Steuerdaten enthält, die von
der Funktion (78, 120) interpretiert werden sollen.
45. System nach Anspruch 44,
dadurch gekennzeichnet, dass
das Steuerfeld (60) wenigstens einen Modulanzeiger (80)
umfasst, der das nächste Analysemodul anzeigt, welches
ausgeführt werden soll.
46. System nach Anspruch 44,
dadurch gekennzeichnet, dass
das Steuerfeld (60) wenigstens eine
Parameteridentifikation (74) umfasst, die die Stelle
eines Parameters (46) in einem Speicher (116)
identifiziert.
47. System nach Anspruch 46,
dadurch gekennzeichnet, dass
die Funktion (78, 120) den Parameterwert analysiert, der
in der Speicherstelle gespeichert ist.

48. System nach Anspruch 46,
dadurch gekennzeichnet, dass
das Steuerfeld (60) ferner wenigstens einen vorgegebenen
Wert umfasst und die Funktion (78, 120) den vorgegebenen
Wert in die Speicherstelle schreibt.
49. System zum Kompilieren und Ausführen wenigstens einer
Analyse, welches zum Steuern eines industriellen
Prozesses verwendet wird, umfassend:
- eine Einrichtung (40) zum Bereitstellen einer Vielzahl
von High-Level-Analyseprimitiven (78, 120), die jeweils
eine Bildungs-Prozedur (114) und eine Do-Prozedur (112)
umfassen;
 - eine Einrichtung (102) zum Kompilieren einer Vielzahl von
Steuerebenen-Analysemodulen (70, 103) unter Verwendung
der Bildungsprozedur (114) von wenigstens einem der High-
Level-Analyseprimitive (78, 120), wobei die Steuerebenen-
Analysemodule (70, 130) eine Baumstruktur bilden, die
wenigstens eine Analyse darstellt; und
 - eine Einrichtung (108) zum Ausführen jedes Steuerebenen-
Analysemoduls (70, 103) in der Baumstruktur unter
Verwendung der Do-Prozedur (112) des High-Level-
Analyseprimitivs (78, 120), welches zum Kompilieren des
Steuerebenen-Analysemoduls (70, 103) verwendet wurde.
50. System nach Anspruch 49, umfassend:
- eine Einrichtung zum Bereitstellen einer Vielzahl von
Low-Level-Analyseprimitiven (120a-m);
 - eine Einrichtung zum Kompilieren einer Vielzahl von Low-
Level-Analysemodulen (130a) unter Verwendung der Low-
Level-Analyseprimitive;

eine Einrichtung zum Kompilieren einer Vielzahl von High-Level-Analysemodulen (130b) unter Verwendung der Low-Level-Analysemodule (130a), wobei jedes High-Level-Analysemodul (130b) eine Bildungsprozedur und eine Do-Prozedur umfaßt;

eine Einrichtung zum Kompilieren einer Vielzahl von neuen Steuerebenen-Analysemodulen (130c) unter Verwendung der Bildungsprozedur von wenigstens einem der High-Level-Analysemodule (130b), wobei die neuen Steuerebenen-Analysemodule (130c) zu der Baumstruktur hinzugefügt werden; und

eine Einrichtung (108) zum Ausführen jedes neuen Steuerebenen-Analysemoduls in der Baumstruktur unter Verwendung der Do-Prozedur des High-Level-Analysemoduls (130b), das verwendet wurde, um das neue Steuerebenen-Analysemodul (130c) zu kompilieren.

51. System nach Anspruch 50,
dadurch gekennzeichnet, dass
die High-Level-Analyseprimitive (78, 120) und die Low-Level-Analyseprimitive (120a-m) eingebaute Funktionen sind.
52. System nach Anspruch 50,
dadurch gekennzeichnet, dass
die High-Level-Analysemodule (130b) und die Low-Level-Analysemodule (130a) benutzerdefinierte Funktionen sind.

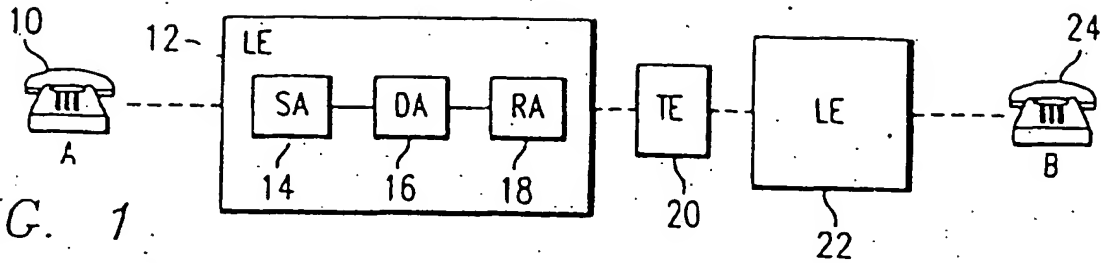


FIG. 1

(Stand der Technik)

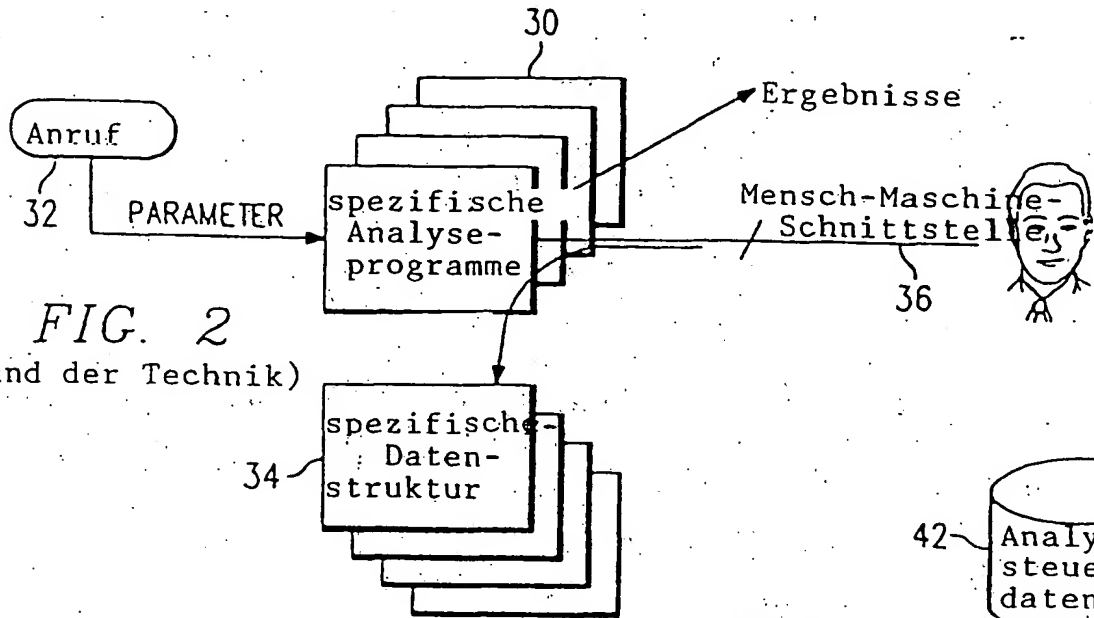


FIG. 2

(Stand der Technik)

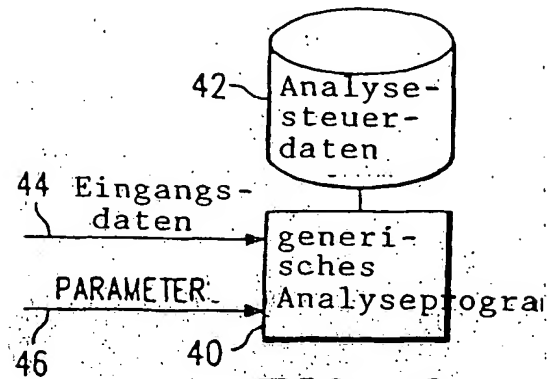


FIG. 3

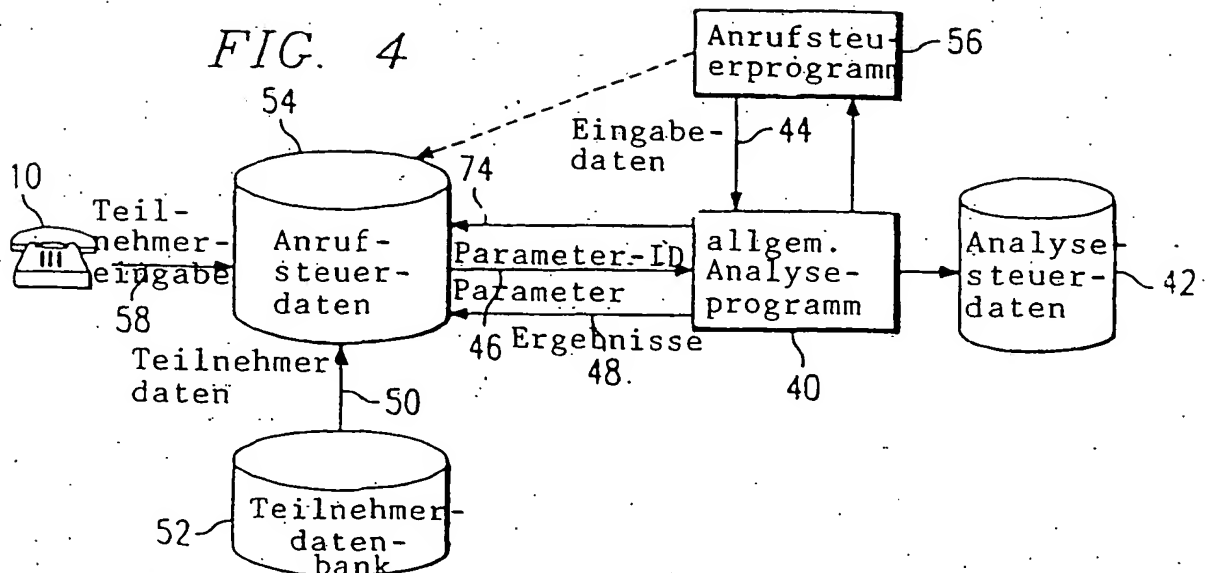


FIG. 4

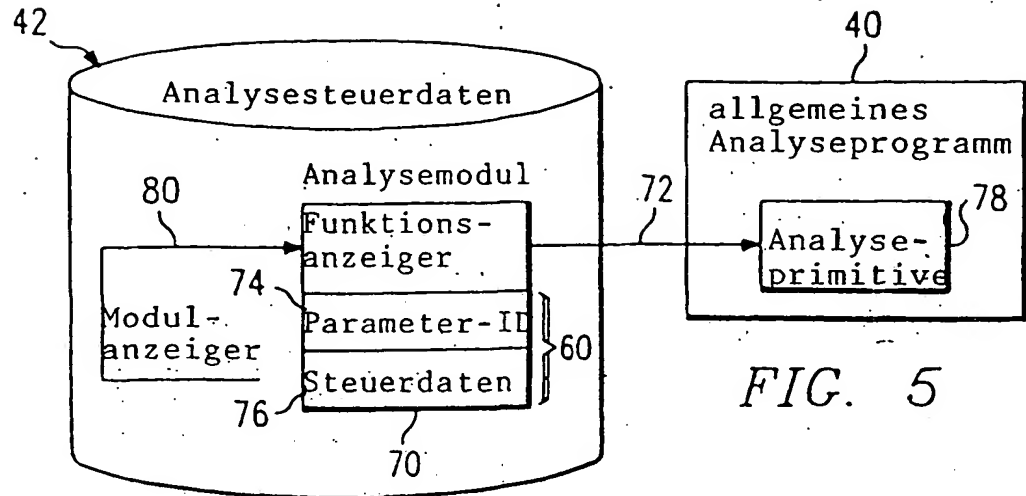


FIG. 5

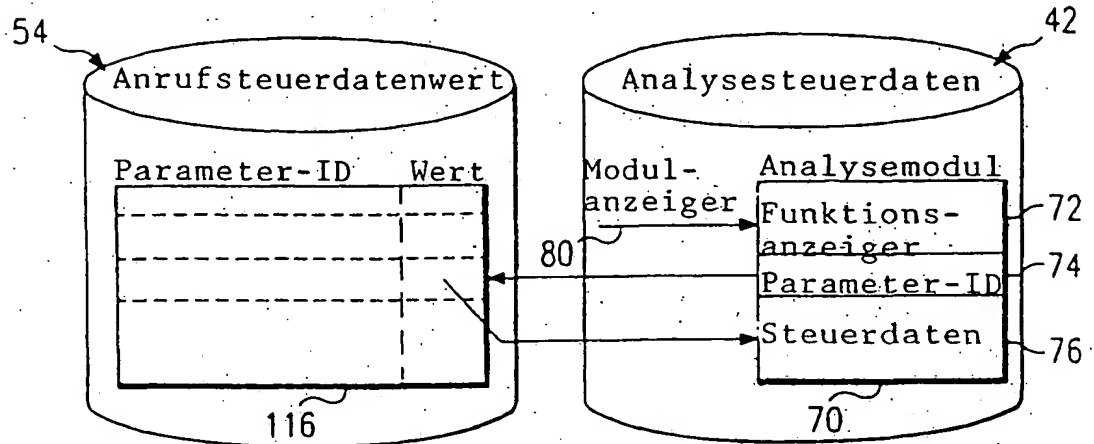


FIG. 6

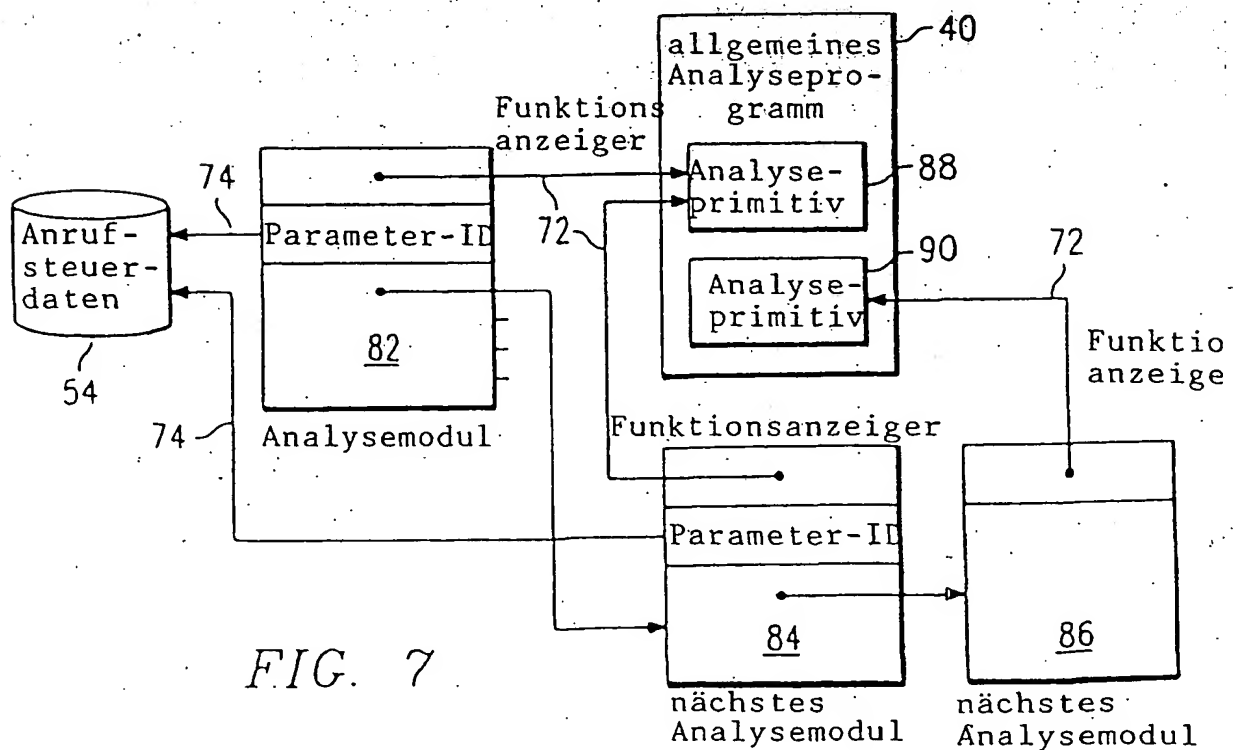


FIG. 7

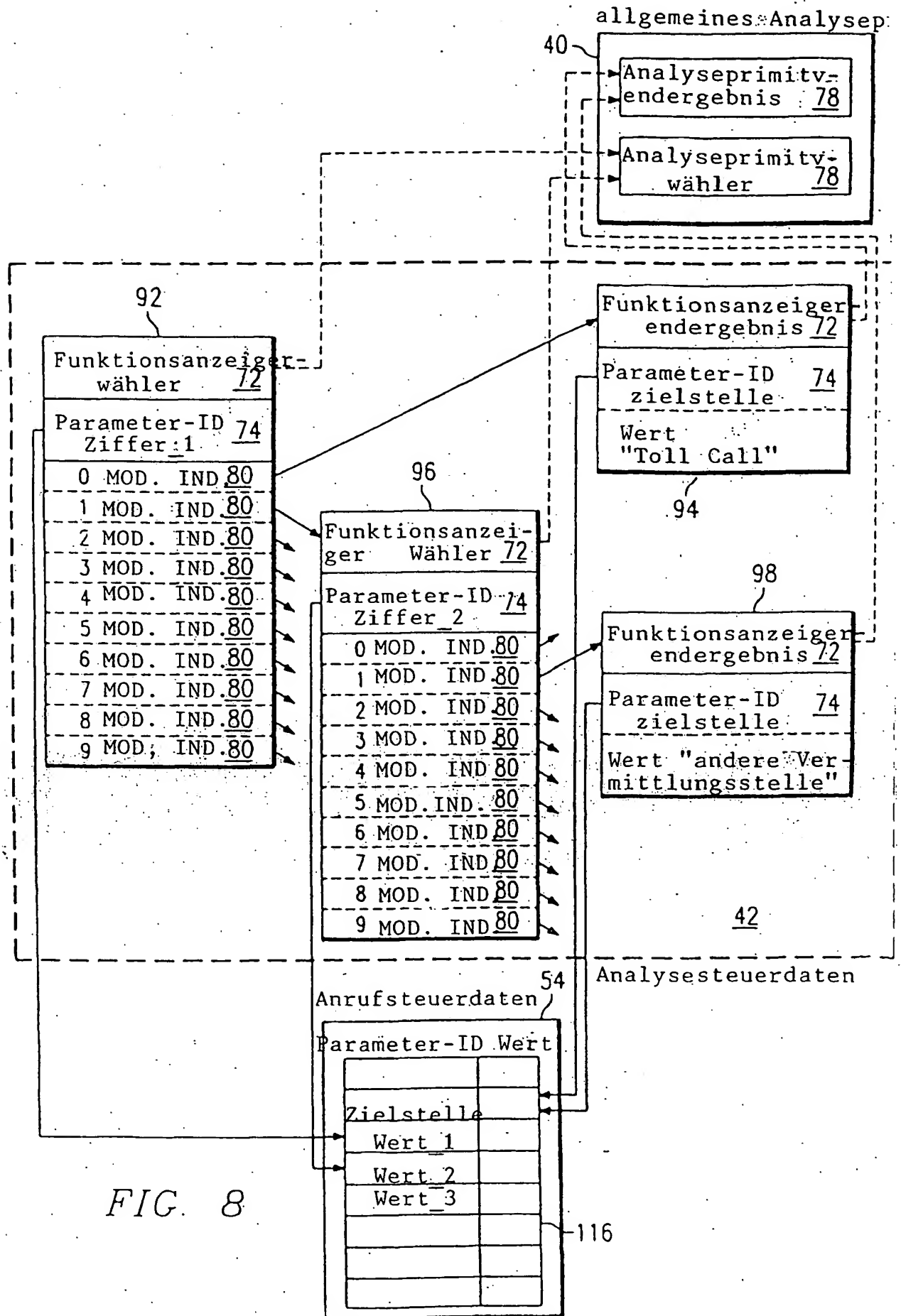


FIG. 8

FIG. 9A

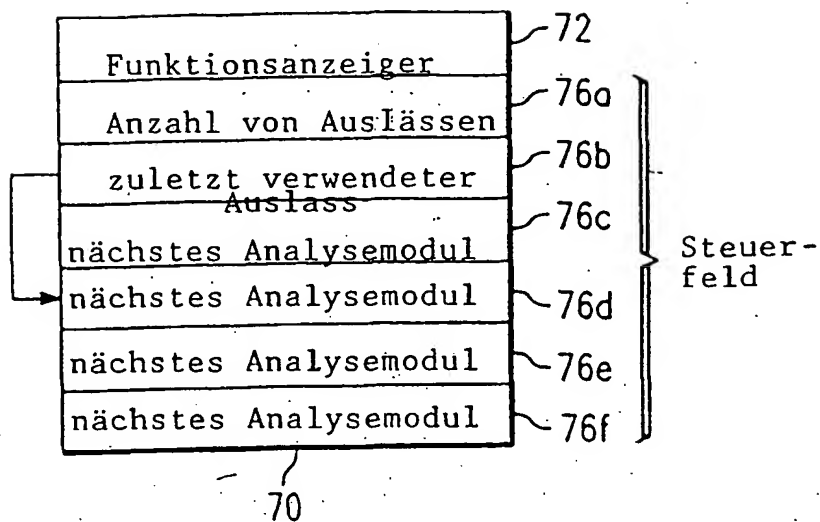
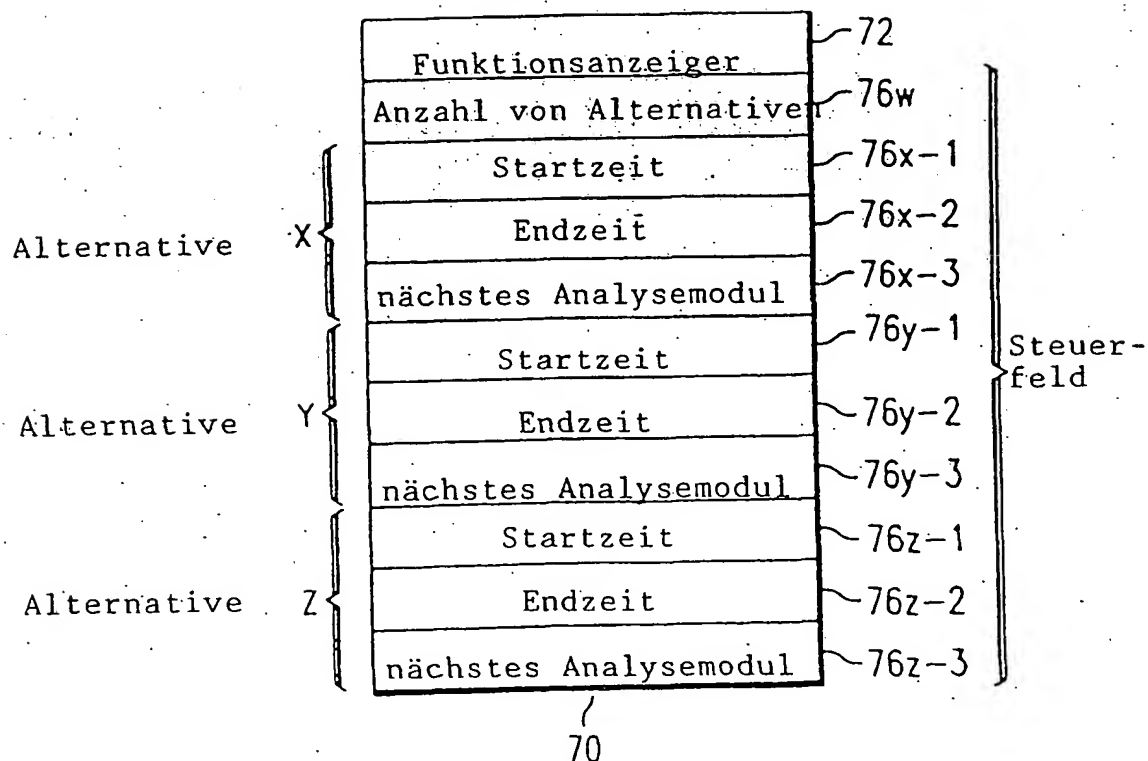
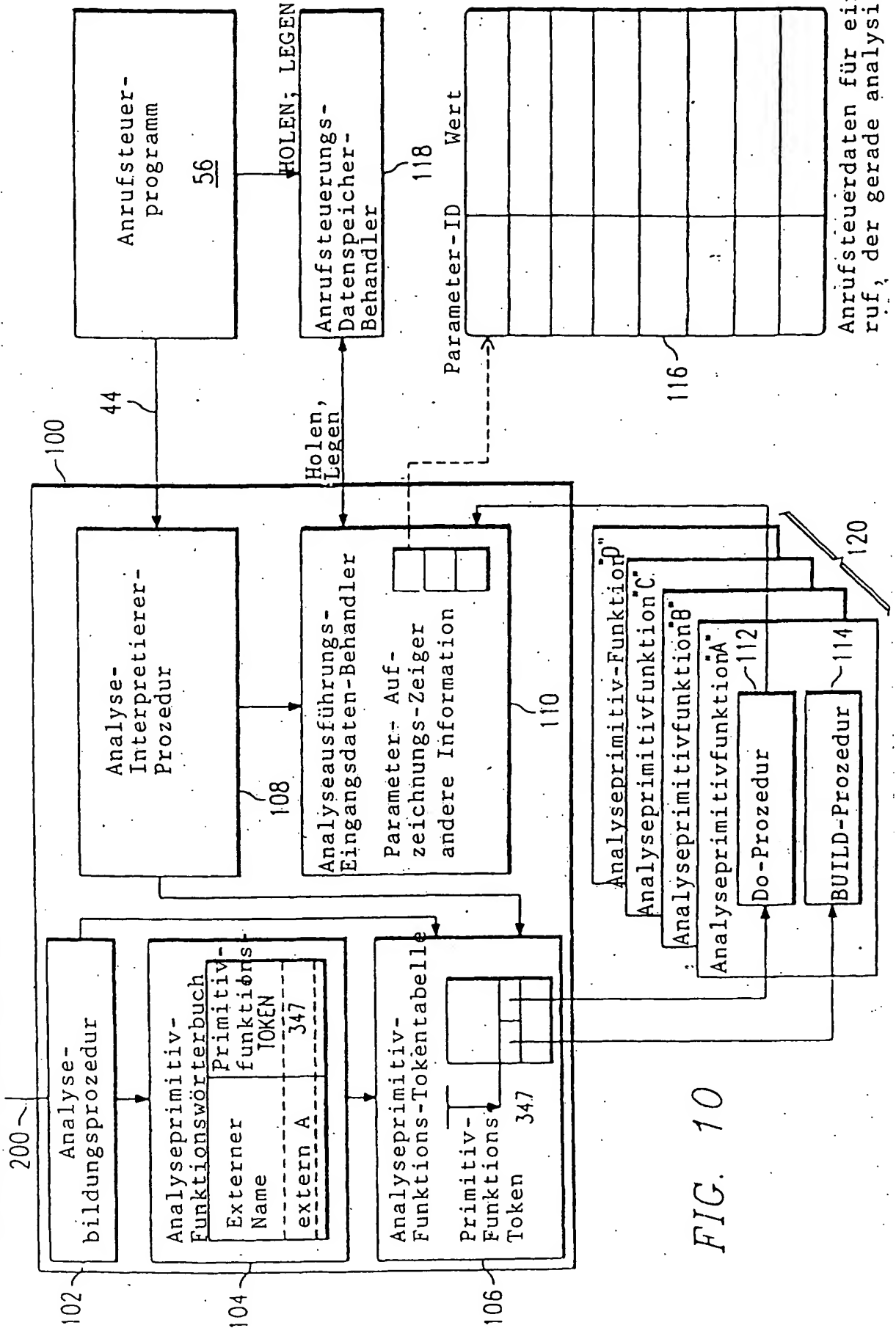


FIG. 9B



Betreiberschnittstelle



Anrufsteuerdaten für einen Anruf, der gerade analysiert

FIG. 10

FIG. 11

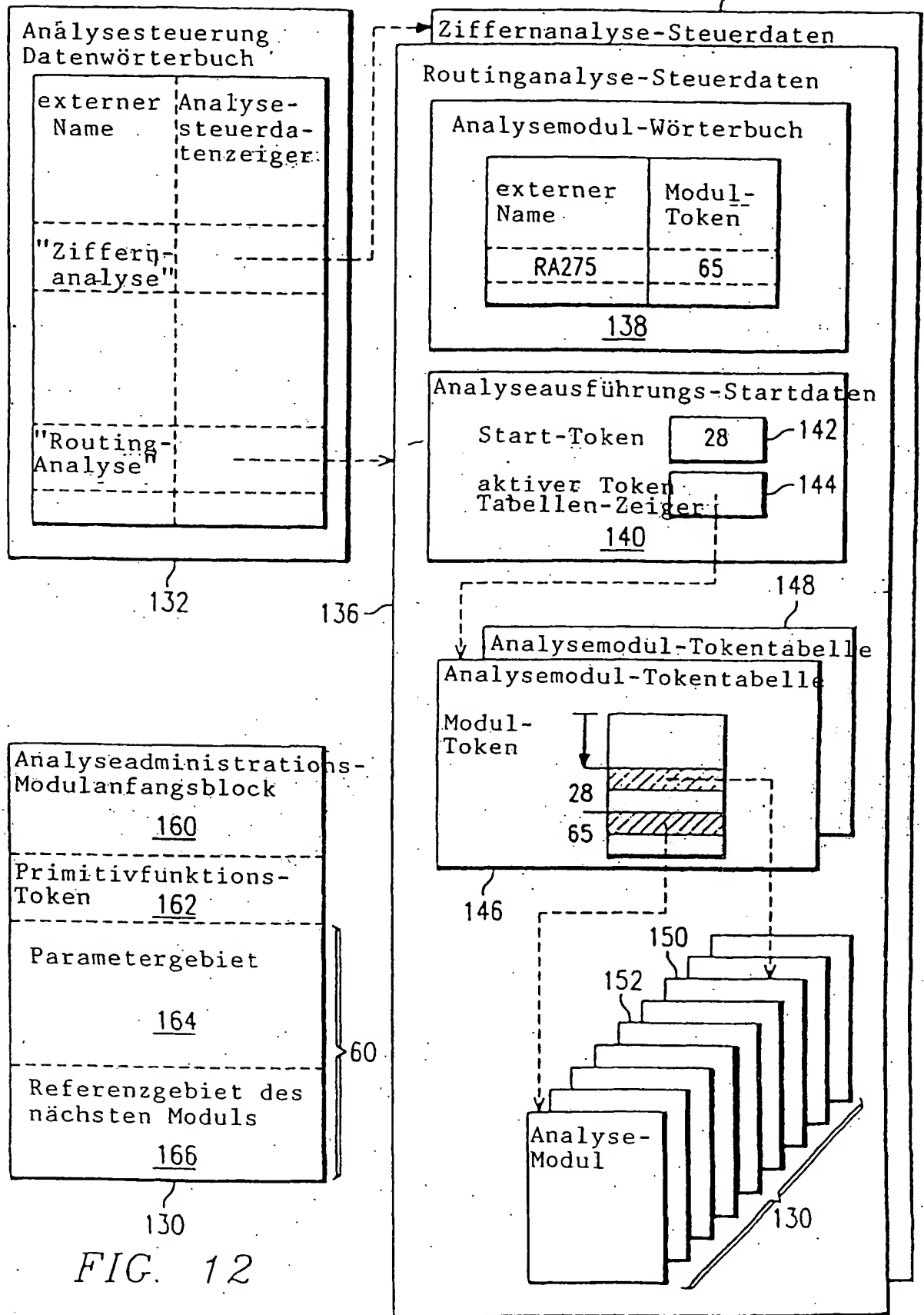


FIG. 12

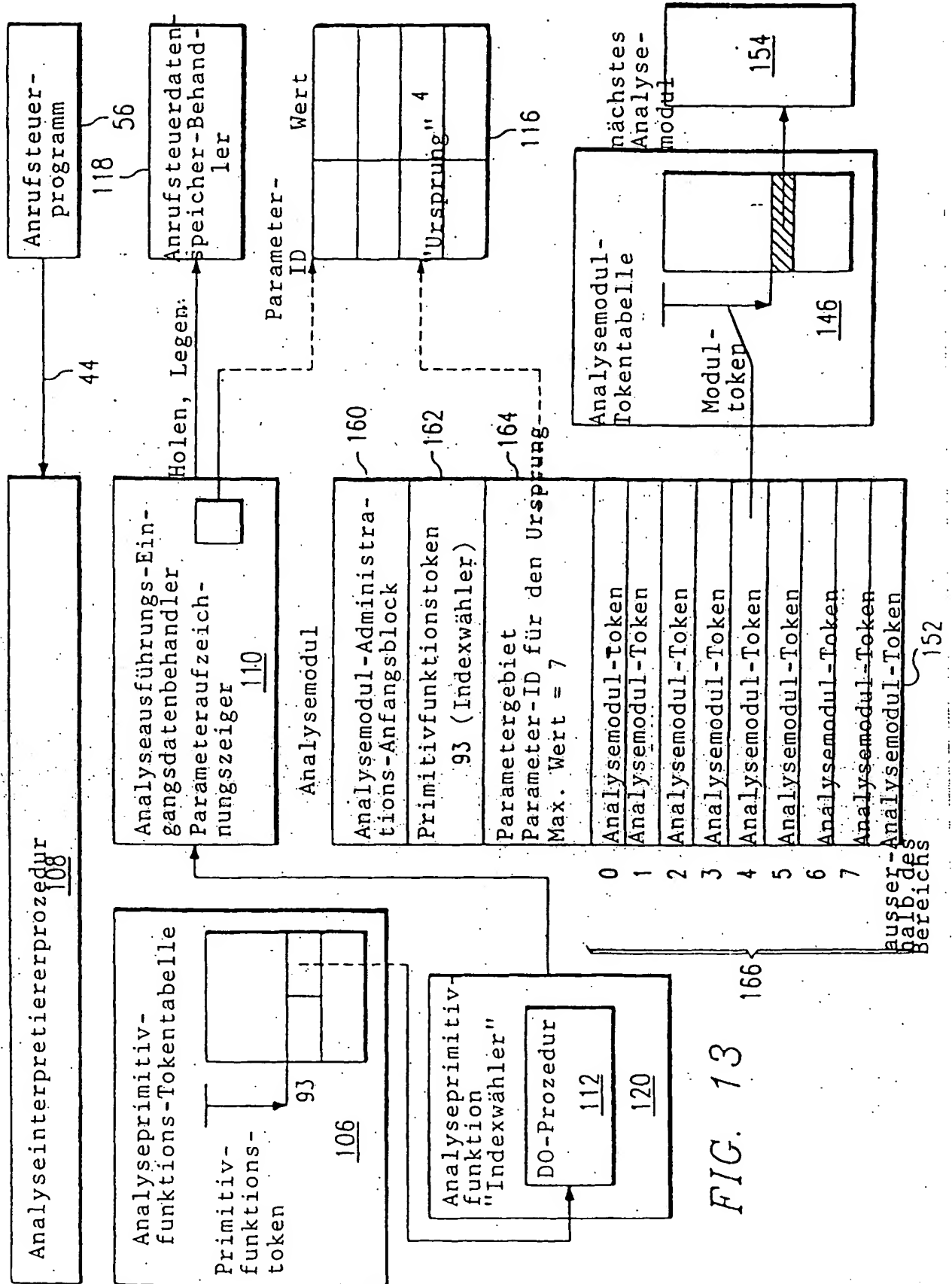


FIG. 14

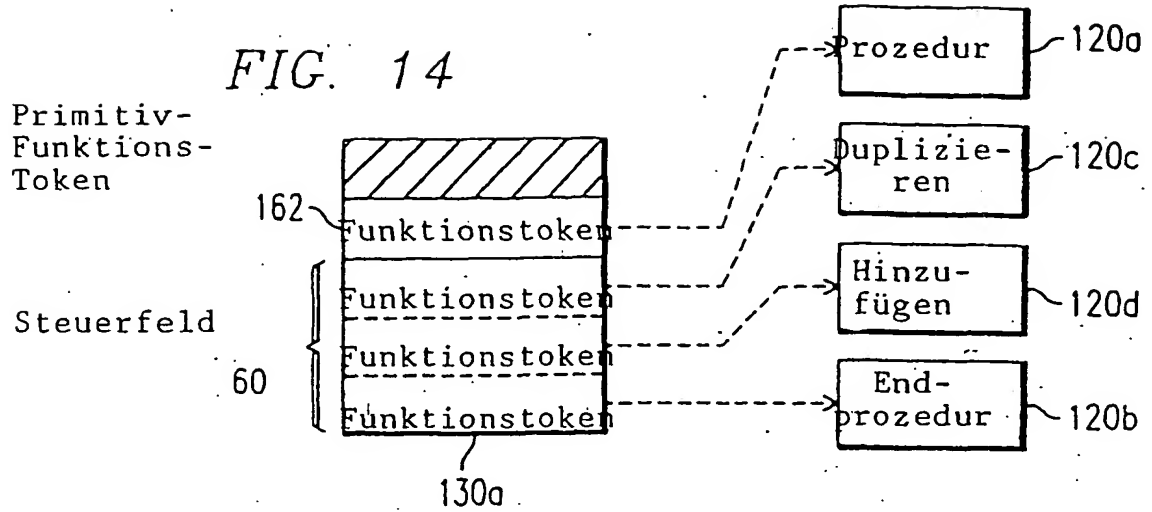


FIG. 16

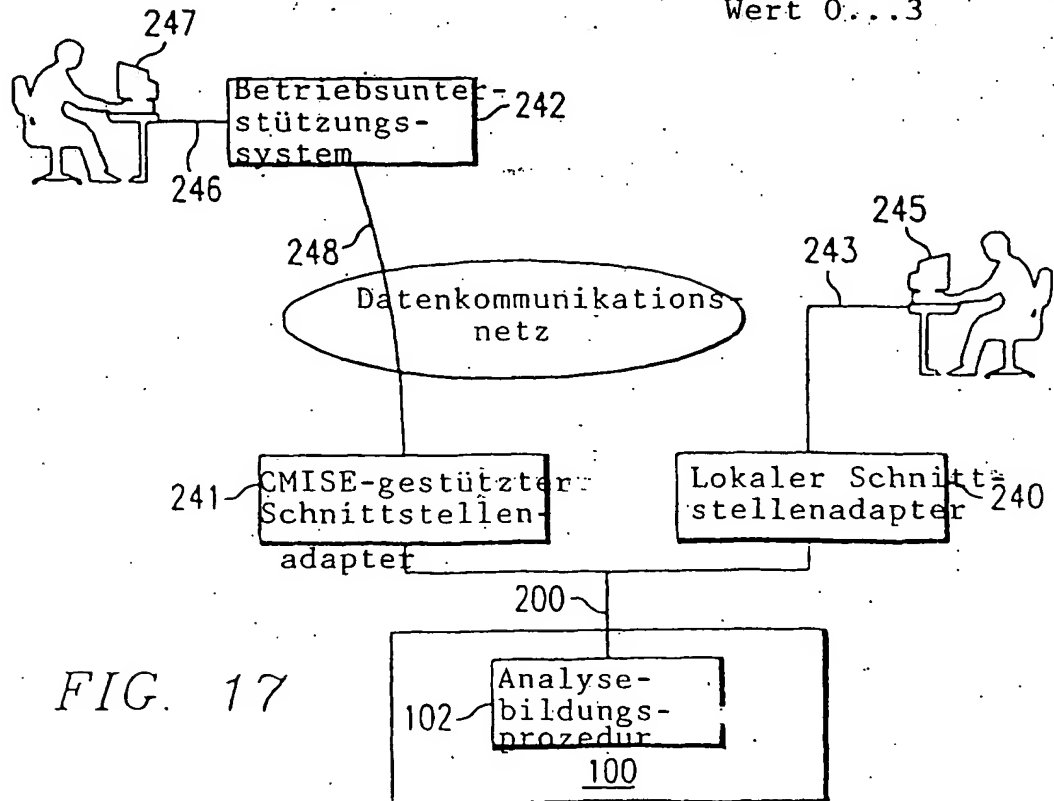
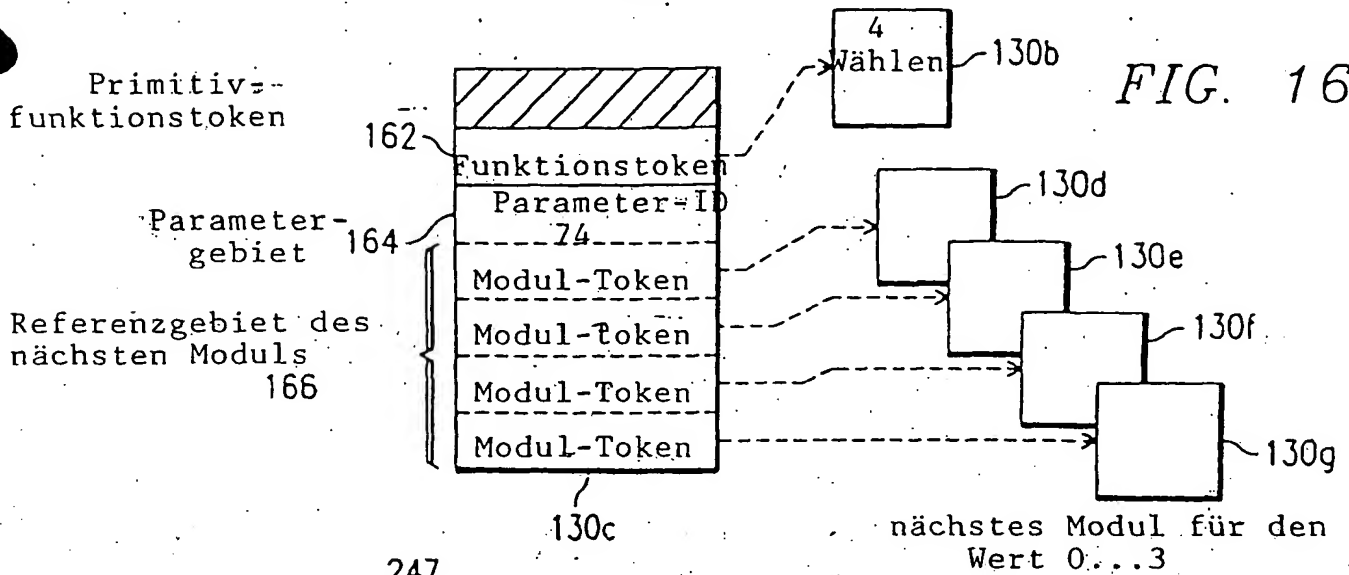
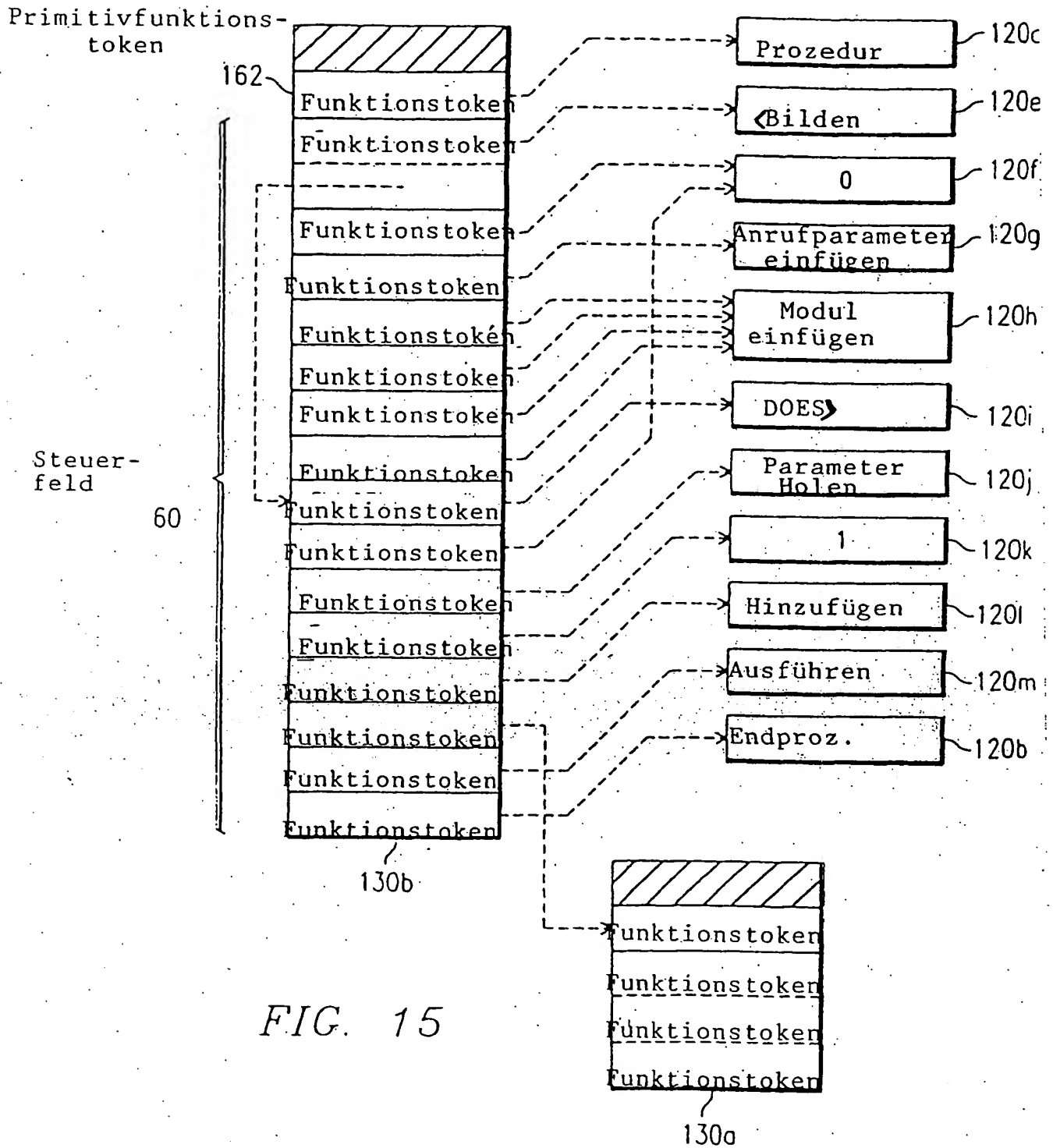


FIG. 17



**This Page is Inserted by IFW Indexing and Scanning
Operations and is not part of the Official Record**

BEST AVAILABLE IMAGES

Defective images within this document are accurate representations of the original documents submitted by the applicant.

Defects in the images include but are not limited to the items checked:

- ☒ BLACK BORDERS
- ☐ IMAGE CUT OFF AT TOP, BOTTOM OR SIDES
- ☐ FADED TEXT OR DRAWING
- ☐ BLURRED OR ILLEGIBLE TEXT OR DRAWING
- ☐ SKEWED/SLANTED IMAGES
- ☐ COLOR OR BLACK AND WHITE PHOTOGRAPHS
- ☐ GRAY SCALE DOCUMENTS
- ☐ LINES OR MARKS ON ORIGINAL DOCUMENT
- ☐ REFERENCE(S) OR EXHIBIT(S) SUBMITTED ARE POOR QUALITY
- ☐ OTHER: _____

IMAGES ARE BEST AVAILABLE COPY.

As rescanning these documents will not correct the image problems checked, please do not report these problems to the IFW Image Problem Mailbox.